aws

Panduan Pengguna

# Amazon Elastic Compute Cloud



Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

# Amazon Elastic Compute Cloud: Panduan Pengguna

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan kekayaan masing-masing pemiliknya, yang mungkin berafiliasi, terkait dengan, atau disponsori oleh Amazon, atau mungkin tidak.

# Table of Contents

Apa itu Amazon EC2? ..... 1
Fitur ..... 1
Layanan terkait ..... 2
Mengakses EC2 ..... 4
Harga ..... 5
Estimasi, penagihan, dan optimalisasi biaya ..... 6
Sumber daya ..... 7
Mulai tutorial ..... 8
Langkah 1: Luncurkan instans ..... 10
Langkah 2: Hubungkan ke instans Anda ..... 12
Langkah 3: Bersihkan instans Anda ..... 15
Langkah selanjutnya ..... 16
Praktik terbaik ..... 17
Amazon Machine Image ..... 20
Karakteristik AMI ..... 22
Izin peluncuran ..... 22
Tipe volume root ..... 23
Mengidentifikasi jenis volume root AMI ..... 24
Tipe virtualisasi ..... 26
Temukan AMI ..... 28
Parameter Manajer Sistem ..... 31
Parameter publik Manajer Sistem ..... 35
AMI berbayar di AWS Marketplace ..... 39
Jual AMI Anda di AWS Marketplace ..... 40
Mencari AMI berbayar ..... 40
Membeli AMI berbayar ..... 42
Ambil kode produk ..... 43
Menggunakan dukungan berbayar ..... 44
Tagihan untuk AMI berbayar dan didukung ..... 45
Mengelola langganan Anda ..... 45
Siklus hidup AMI ..... 46
Membuat AMI ..... 48
Buat S3-backed AMI Amazon ..... 57
Buat AMI menggunakan Windows Sysprep ..... 97

Menyalin AMI ..... 115
Menyimpan dan memulihkan AMI ..... 133
Nenek moyang AMI ..... 143
Tanda air AMI ..... 148
Jenis instans yang diizinkan AMI ..... 155
Penggunaan AMI ..... 160
Usangkan AMI ..... 185
Menonaktifkan AMI ..... 191
Membatalkan pendaftaran AMI ..... 197
Mode boot ..... 204
Persyaratan untuk mode boot UEFI ..... 205
Parameter mode boot AMI ..... 207
Mode boot tipe instans ..... 209
Mode boot instans ..... 212
Mode boot sistem operasi ..... 214
Mengatur mode boot AMI ..... 215
Variabel UEFI ..... 222
UEFI Secure Boot ..... 223
Enkripsi AMI ..... 240
Instance-launching skenario ..... 240
Image-copying skenario ..... 244
AMI Bersama ..... 246
Penyedia AMI terverifikasi ..... 247
Mencari AMI bersama ..... 248
Bersiaplah untuk menggunakan AMI bersama untuk Linux ..... 251
AMI yang diizinkan ..... 252
Jadikan AMI Anda publik ..... 274
Blokir akses publik untuk AMI ..... 277
Bagikan AMI dengan organisasi dan unit organisasi ..... 287
Bagikan AMI dengan spesifik AWS rekening ..... 299
Membatalkan berbagi AMI dengan akun Anda ..... 303
Rekomendasi untuk membuat AMI Linux bersama ..... 305
Memantau peristiwa AMI ..... 311
Detail peristiwa ..... 312
tersedia peristiwa ..... 313
gagal peristiwa ..... 313

dibatalkan pendaftaran peristiwa ... 314
dinonaktifkan peristiwa ... 315
Memahami penagihan AMI ... 315
Bidang penagihan AMI ... 316
Mencari informasi penagihan AMI ... 318
Memverifikasi biaya AMI pada tagihan Anda ... 320
Kuota AMI ... 321
Minta kenaikan kuota untuk AMIs ... 322
Instans ... 324
Tipe instans ... 325
Tipe instans yang tersedia ... 326
Spesifikasi perangkat keras ... 327
Jenis hypervisor ... 327
Tipe virtualisasi AMI ... 328
Prosesor ... 328
Menemukan tipe instans ... 331
Pencari tipe instans EC2 ... 339
Rekomendasi Compute Optimizer ... 341
Perubahan jenis instans ... 344
Instance performa yang dapat melonjak ... 353
Instans GPU ... 407
Instans Mac ... 421
Optimisasi EBS ... 477
Opsi CPU ... 609
AMD SEV-SNP ... 921
Kontrol status prosesor ... 936
Instans terkelola ... 941
Penagihan untuk instans terkelola ... 942
Mengidentifikasi instans terkelola ... 942
Pengaturan visibilitas sumber daya terkelola ... 944
Memulai dengan instans terkelola ... 949
Virtualisasi Bersarang ... 949
Cara kerjanya ... 950
Pertimbangan-pertimbangan ... 950
Luncurkan instance baru dengan virtualisasi bersarang diaktifkan ... 951
Mengkonfigurasi instance yang ada untuk menggunakan virtualisasi bersarang ... 952

Temukan jenis instans yang mendukung virtualisasi bersarang ..... 953
Opsi penagihan dan pembelian ..... 954
Instans Sesuai Permintaan ..... 955
Instans Terpesan ..... 958
Instans Spot ..... 1028
Host Khusus ..... 1131
Instans Khusus ..... 1193
Reservasi Kapasitas ..... 1201
Templat peluncuran ..... 1360
Pembatasan ..... 1361
Izin ..... 1362
Kontrol instans peluncuran ..... 1370
Buat ..... 1372
Modifikasi (kelola versi) ..... 1388
Delete ..... 1393
Luncurkan sebuah instans ..... 1395
Tutorial ..... 1398
Referensi parameter instance ..... 1424
Luncurkan menggunakan wizard peluncuran instans ..... 1439
Meluncurkan menggunakan templat peluncuran ..... 1442
Luncurkan dari instance yang sudah ada ..... 1449
Peluncuran dari AWS Marketplace AMI ..... 1451
Terhubung ke Instans Anda ..... 1454
Prasyarat koneksi umum ..... 1455
Hubungkan ke instans Linux Anda, menggunakan SSH. ..... 1461
Hubungkan ke instans Windows Anda menggunakan RDP ..... 1478
Terhubung menggunakan Session Manager ..... 1488
Hubungkan menggunakan IP publik dan EC2 Instance Connect ..... 1489
Menghubungkan menggunakan IP pribadi dan EC2 Instance Connect Endpoint ..... 1526
Perubahan status instans ..... 1559
Penagihan berdasarkan status instans ..... 1560
Instans yang tertunda ..... 1561
Instans yang dihentikan ..... 1562
Instans hibernasi ..... 1563
Mem-boot ulang instance ..... 1563
Instans yang dihentikan ..... 1564

Perbedaan antara status instance ..... 1564
Berhenti dan mulai ..... 1567
Hibernasi ..... 1587
Mulai ulang ..... 1617
Mengakhiri ..... 1619
Pensiun ..... 1637
Pemulihan instans otomatis ..... 1642
Konsep kunci pemulihan instans otomatis ..... 1644
Perbedaan antara pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatch tindakan ..... 1646
Membangun sistem yang tangguh ..... 1647
Verifikasi apakah pemulihan otomatis terjadi ..... 1647
Pemulihan otomatis yang disederhanakan ..... 1649
CloudWatch pemulihan berbasis tindakan ..... 1654
Metadata instans ..... 1658
Kategori metadata instans ..... 1660
Kategori data dinamis ..... 1675
Akses metadata instans ..... 1676
Konfigurasikan opsi IMDS ..... 1718
Menjalankan perintah saat peluncuran ..... 1756
Contoh: Nilai indeks peluncuran AMI ..... 1782
Mendeteksi apakah host adalah instance EC2 ..... 1787
Memeriksa dokumen identitas instans ..... 1787
Periksa sistem UUID ..... 1787
Periksa pengenal pembuatan mesin virtual sistem ..... 1789
Dokumen identitas instans ..... 1794
Mengambil dokumen identitas instance ..... 1796
Verifikasi dokumen identitas instans ..... 1797
Sertifikat publik ..... 1809
Kepatuhan STIG ..... 1871
Pengaturan STIG ..... 1873
Unduh skrip STIG ..... 1896
Terapkan pengaturan STIG dengan Systems Manager ..... 1918
Sinkronisasi jam ..... 1921
Stempel waktu paket perangkat keras ..... 1922
Detik kabisat ..... 1923

Gunakan Layanan Sinkronisasi Waktu Amazon lokal ..... 1924
Gunakan Layanan Sinkronisasi Waktu Amazon publik ..... 1941
Bandingkan stempel waktu untuk instans Linux Anda ..... 1943
Mengubah zona waktu instans Anda ..... 1945
Manajer Kapasitas EC2 ..... 1947
Fitur Utama ..... 1948
Mengaktifkan Manajer Kapasitas ..... 1949
Mengatur data Anda ..... 1962
Menavigasi Manajer Kapasitas di konsol ..... 1980
Mengekspor data Anda ..... 1982
Kelola driver perangkat ..... 1987
Driver jaringan ..... 1987
Driver grafis ..... 1988
Driver perangkat penyimpanan ..... 1988
Driver perangkat Windows lainnya ..... 1988
Driver AMD ..... 1988
Driver NVIDIA ..... 1995
Instal driver ENA di Windows ..... 2043
Driver Windows PV ..... 2064
NVMe driver ..... 2100
Driver perangkat Windows lainnya ..... 2112
Konfigurasikan instance Windows ..... 2113
Windows-specific pengaturan sistem ..... 2113
AWS driver perangkat untuk instans Windows ..... 2114
Agen peluncuran Windows ..... 2116
Peluncuran Cepat EC2 untuk Windows ..... 2277
Ubah kata sandi Administrator Windows ..... 2303
Tambahkan komponen Sistem Windows ..... 2304
Instal WSL di Windows ..... 2309
Utilitas Windows ..... 2311
Mutakhirkan instans Windows ..... 2313
Lakukan pemutakhiran langsung ..... 2314
Lakukan pemutakhiran otomatis ..... 2318
Bermigrasi ke jenis Nitro-based instance ..... 2330
Memecahkan masalah pemutakhiran ..... 2339
Tutorial: Hubungkan instance EC2 ke database RDS ..... 2340

Tujuan Tutorial ..... 2340
Konteks ..... 2340
Arsitektur ..... 2341
Pertimbangan-pertimbangan ..... 2342
Waktu untuk menyelesaikan tutorial ..... 2343
Biaya ..... 2343
Opsi 1: Secara otomatis terhubung menggunakan konsol EC2 ..... 2344
Opsi 2: Secara otomatis terhubung menggunakan konsol RDS ..... 2356
Opsi 3: Hubungkan secara manual ..... 2368
Armada ..... 2379
Fitur dan manfaat ..... 2379
Metode armada mana yang digunakan? ..... 2380
Opsi konfigurasi ..... 2382
Tipe permintaan ..... 2384
Batas pengeluaran ..... 2413
Attribute-based pemilihan jenis contoh ..... 2416
Pembobotan instans ..... 2453
Strategi alokasi ..... 2455
Penyeimbangan Ulang Kapasitas ..... 2463
Reservasi Kapasitas ..... 2469
Bekerja dengan EC2 Fleet ..... 2471
Status permintaan Armada EC2 ..... 2472
Prasyarat Armada EC2 ..... 2473
Membuat Armada EC2 ..... 2477
Menandai Armada EC2 ..... 2487
Jelaskan Armada EC2 ..... 2490
Memodifikasi Armada EC2 ..... 2494
Hapus Armada EC2 ..... 2496
Bekerja dengan Spot Fleet ..... 2501
Status permintaan Armada Spot ..... 2503
Izin Armada Spot ..... 2504
Membuat Armada Spot ..... 2515
Menandai Armada Spot ..... 2524
Jelaskan Armada Spot ..... 2534
Memodifikasi permintaan Armada Spot ..... 2536
Membatalkan (menghapus) permintaan Spot Fleet ..... 2538

Penskalaan otomatis untuk Armada Spot ..... 2541
Pantau armada Anda ..... 2552
Memantau armada Anda menggunakan CloudWatch ..... 2553
Memantau armada Anda menggunakan EventBridge ..... 2556
Tutorial ..... 2574
Tutorial: Mengkonfigurasi Armada EC2 untuk menggunakan pembobotan instans ..... 2577
Tutorial: Mengkonfigurasi Armada EC2 untuk menggunakan On-Demand Instans sebagai kapasitas utama ..... 2581
Tutorial: Mengkonfigurasi Armada EC2 untuk meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan ..... 2582
Tutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Blok Kapasitas 2589
Tutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Reservasi Kapasitas yang Dapat Diinterupsi ..... 2592
Tutorial: Mengkonfigurasi Armada EC2 Anda untuk meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas ..... 2597
Contoh konfigurasi CLI untuk Armada EC2 ..... 2603
Contoh 1: Meluncurkan Instans Spot sebagai opsi pembelian default ..... 2603
Contoh 2: Luncur On-Demand kan Instans sebagai opsi pembelian default ..... 2604
Contoh 3: Luncur On-Demand kan Instans sebagai kapasitas utama ..... 2605
Contoh 4: Luncurkan On-Demand Instans menggunakan beberapa Reservasi Kapasitas .. 2605
Contoh 5: Luncur On-Demand kan Instans menggunakan Reservasi Kapasitas ketika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak digunakan ..... 2609
Contoh 6: Meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan ..... 2613
Contoh 7: Konfigurasikan Penyeimbangan Kembali Kapasitas untuk meluncurkan Instans Spot pengganti ..... 2616
Contoh 8: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas ..... 2618
Contoh 9: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas ..... 2619
Contoh 10: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas harga ..... 2621
Contoh 11: Mengkonfigurasi pemilihan jenis instans berbasis atribut ..... 2622
Contoh 12: Meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas ..... 2624
Contoh konfigurasi CLI Spot Fleet ..... 2625
Contoh 1: Zona Lowest-priced Ketersediaan ..... 2626
Contoh 2: Zona Keter Lowest-priced sediaan dari daftar ..... 2627

Contoh 3: jenis Lowest-priced instance dari daftar ..... 2629
Contoh 4. Menimpa harga untuk permintaan ..... 2631
Contoh 5: Gunakan strategi alokasi yang beragam ..... 2632
Contoh 6: Gunakan pembobotan instans ..... 2636
Contoh 7: Gunakan On-Demand kapasitas ..... 2637
Contoh 8: Konfigurasikan Penyeimbangan Kembali Kapasitas ..... 2638
Contoh 9: Gunakan armada yang dioptimalkan kapasitas ..... 2640
Contoh 10: Gunakan armada yang dioptimalkan kapasitas dengan prioritas ..... 2641
Contoh 11: Gunakan priceCapacityOptimized armada ..... 2642
Contoh 12: Mengonfigurasi pemilihan tipe instans berbasis atribut ..... 2643
Kuota armada ..... 2644
Meminta peningkatan kuota untuk kapasitas target ..... 2646
Jaringan ..... 2647
Wilayah dan Zona ..... 2648
Wilayah ..... 2649
Zona Ketersediaan ..... 2649
Zona Lokal ..... 2651
Wavelength Zones ..... 2652
AWS Outposts ..... 2653
Pengalamatan IP instans ..... 2655
Alamat IPv4 privat ..... 2656
Alamat IPv4 publik ..... 2657
Pengoptimalan alamat IPv4 publik ..... 2659
Alamat IPv6 ..... 2661
Beberapa alamat IP ..... 2662
Nama host instans EC2 ..... 2663
Link-local alamat ..... 2663
Alamat IPv4 ..... 2664
Alamat IPv6 ..... 2670
Alamat IP sekunder ..... 2676
Alamat IPv4 di Windows ..... 2680
Nama host dan domain instans EC2 ..... 2687
Memahami nama host instans EC2 dan domain ..... 2688
Jenis nama host ..... 2688
Bawa alamat IP Anda sendiri ..... 2703
Definisi BYOIP ..... 2705

Persyaratan dan kuota ..... 2705
Ketersediaan wilayah ..... 2706
Ketersediaan Local Zone ..... 2706
Prasyarat ..... 2709
Masukkan rentang alamat Anda ..... 2717
Gunakan rentang alamat Anda ..... 2726
Alamat IP elastis ..... 2728
Harga alamat IP Elastis ..... 2728
Dasar alamat IP Elastis ..... 2729
Kuota alamat IP Elastis ..... 2730
Kaitkan sebuah alamat IP Elastis ..... 2730
Mentransfer alamat IP Elastic ..... 2736
Merilis alamat IP Elastis ..... 2741
Menggunakan DNS terbalik untuk aplikasi email ..... 2743
Antarmuka jaringan ..... 2747
Konsep antarmuka jaringan ..... 2748
Kartu jaringan ..... 2751
Alamat IP per antarmuka jaringan ..... 2755
Membuat antarmuka jaringan ..... 2758
Lampiran antarmuka jaringan ..... 2763
Mengelola alamat IP ..... 2767
Memodifikasi atribut antarmuka jaringan ..... 2771
Beberapa antarmuka jaringan ..... 2775
Requester-managed antarmuka jaringan ..... 2778
Delegasi awalan ..... 2780
Menghapus antarmuka jaringan ..... 2791
bandwidth jaringan ..... 2792
Bandwidth instans yang tersedia ..... 2793
Pembobotan bandwidth ..... 2795
Memantau bandwidth instans ..... 2835
Jaringan yang ditingkatkan ..... 2835
Adaptor Jaringan Elastis (ENA) ..... 2837
ENA Ekspres ..... 2919
Intel 82599 VF ..... 2960
Memantau kinerja jaringan ..... 2973
Tingkatkan latensi jaringan di Linux ..... 2984

Pertimbangan kinerja nitro ..... 2988
Optimalkan kinerja jaringan pada Windows ..... 2997
Elastic Fabric Adapter ..... 2999
Dasar-dasar EFA ..... 3000
Antarmuka dan pustaka yang didukung ..... 3003
Tipe instans yang didukung ..... 3003
Sistem operasi yang didukung ..... 3020
Batasan EFA ..... 3021
Harga EFA ..... 3022
Memulai dengan EFA dan MPI ..... 3022
Memulai dengan EFA dan NCCL ..... 3041
Memulai dengan EFA dan NIXL ..... 3068
Maksimalkan bandwidth jaringan ..... 3090
Instal EFA dalam wadah ..... 3105
Buat dan lampirkan EFA ..... 3107
Lepaskan dan hapus EFA ..... 3112
Memantau EFA ..... 3112
Verifikasi penginstal EFA ..... 3120
Catatan rilis ..... 3135
Topologi EC2 ..... 3161
Cara kerjanya ..... 3162
Prasyarat ..... 3169
Contoh ..... 3171
Grup penempatan ..... 3186
Strategi penempatan ..... 3188
Buat grup penempatan ..... 3195
Ubah penempatan instans ..... 3197
Menghapus grup penempatan ..... 3199
Grup penempatan bersama ..... 3200
Grup penempatan pada AWS Outposts ..... 3203
MTU Network ..... 3204
Frame jumbo (9001 MTU) ..... 3205
Path MTU Discovery ..... 3206
Mengatur MTU untuk instans Anda ..... 3207
Pemecahan Masalah ..... 3213
Virtual private cloud ..... 3213

VPC default Anda ... 3214
VPC nondefault ... 3215
Akses internet ... 3215
Subnet bersama ... 3216
IPv6-only subnet ... 3216
Jaringan Sekunder ... 3216
Apa itu Jaringan Sekunder? ... 3217
Konsep Utama ... 3218
Arsitektur ... 3218
Pertimbangan tambahan ... 3218
Memulai ... 3219
Mengelola sumber daya Jaringan Sekunder ... 3221
Praktik terbaik desain jaringan ... 3222
Pemecahan masalah ... 3222
Kuota dan batas ... 3223
Keamanan ... 3225
Perlindungan data ... 3226
Keamanan data Amazon EBS ... 3227
Enkripsi saat tidak aktif ... 3228
Enkripsi saat bergerak ... 3229
Keamanan infrastruktur ... 3231
Isolasi jaringan ... 3231
Isolasi pada host fisik ... 3232
Mengontrol lalu lintas jaringan ... 3232
Ketahanan ... 3235
Validasi kepatuhan ... 3236
Manajemen identitas dan akses ... 3236
Identity-based kebijakan ... 3237
Contoh kebijakan untuk API ... 3250
Kebijakan contoh untuk konsol ... 3296
AWS kebijakan yang dikelola ... 3310
Peran IAM ... 3318
Manajemen pembaruan ... 3331
Praktik terbaik untuk instans Windows ... 3332
High-level praktik terbaik keamanan ... 3332
Manajemen pembaruan ... 3333

Manajemen konfigurasi ..... 3336
Manajemen perubahan ..... 3337
Audit dan akuntabilitas untuk instans Windows Amazon EC2 ..... 3337
Pasangan kunci ..... 3338
Membuat pasangan kunci ..... 3340
Jelaskan pasangan kunci Anda ..... 3347
Menghapus pasangan kunci Anda ..... 3352
Menambahkan atau mengganti kunci publik pada instans Linux Anda ..... 3353
Verifikasi sidik jari ..... 3355
Grup keamanan ..... 3358
Gambaran umum ..... 3359
Membuat grup keamanan ..... 3360
Mengubah grup keamanan untuk instance Anda ..... 3362
Menghapus grup keamanan ..... 3369
Pelacakan koneksi ..... 3370
Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda ..... 3376
NitroTPM ..... 3384
Persyaratan ..... 3385
Aktifkan AMI Linux untuk NitroTPM ..... 3386
Verifikasi bahwa AMI diaktifkan untuk NitroTPM ..... 3388
Mengaktifkan atau berhenti menggunakan NitroTPM ..... 3390
Verifikasi bahwa instance diaktifkan untuk NitroTPM ..... 3390
Mengambil kunci dukungan publik ..... 3392
Pengesahan instans EC2 ..... 3394
Integrasi dengan AWS KMS ..... 3396
Membuktikan lingkungan komputasi terisolasi ..... 3396
AWS model tanggung jawab bersama ..... 3397
AMI yang dapat dibuktikan ..... 3397
Persiapkan AWS KMS untuk pengesahan ..... 3407
Dapatkan Dokumen Pengesahan NitroTPM ..... 3408
Mengintegrasikan dengan AWS KMS ..... 3416
Isolasi data dari operator Anda sendiri ..... 3416
Credential Guard untuk instans Windows ..... 3418
Prasyarat ..... 3418
Luncurkan instance yang didukung ..... 3419
Nonaktifkan integritas memori ..... 3420

Aktifkan Credential Guard ..... 3421
Verifikasi bahwa Credential Guard sedang berjalan ..... 3423
AWS PrivateLink ..... 3424
Membuat titik akhir VPC antarmuka ..... 3425
Membuat kebijakan titik akhir ..... 3425
Penyimpanan ..... 3427
AWS Harga penyimpanan ..... 3428
Amazon EBS ..... 3428
Batas volume EBS ..... 3429
Kartu EBS ..... 3433
Penyimpanan instans Amazon EC2 ..... 3436
Persistensi data ..... 3437
Batas volume penyimpanan instans ..... 3439
Volume penyimpanan instans SSD ..... 3443
Menambahkan volume penyimpanan instans ..... 3448
Aktifkan volume swap untuk instans M1 dan C1 ..... 3455
Inisialisasi volume penyimpanan instans ..... 3459
Volume penyimpanan instans statistik kinerja terperinci ..... 3460
Volume akar ..... 3464
Instans dengan volume root Amazon EBS ..... 3465
Instans dengan instance menyimpan volume root (hanya instans Linux) ..... 3467
Pertahankan volume root setelah penghentian instance ..... 3468
Mengganti volume root ..... 3472
Nama perangkat untuk volume ..... 3486
Nama perangkat yang tersedia ..... 3486
Pertimbangan nama perangkat ..... 3489
Pemetaan perangkat blok ..... 3491
Konsep pemetaan perangkat blok ..... 3491
Tambahkan pemetaan perangkat blok ke AMI ..... 3496
Tambahkan pemetaan perangkat blok ke instance ..... 3500
Bagaimana volume dilampirkan dan dipetakan untuk instans Windows ..... 3509
Petakan disk NVME ke volume ..... 3510
Memetakan disk non-NVMe ke volume ..... 3515
Pencegahan tumpang tindih ..... 3525
Ukuran blok yang didukung ..... 3526
Persyaratan ..... 3527

Periksa dukungan instance ... 3528
Konfigurasikan beban kerja ... 3529
Cuplikan Windows VSS EBS ... 3531
Apa itu VSS? ... 3532
Cara kerja solusi snapshot Amazon EBS berbasis VSS ... 3533
Prasyarat VSS ... 3534
Membuat snapshot VSS ... 3547
Memecahkan masalah snapshot VSS ... 3558
Kembalikan opsi untuk solusi AWS VSS ... 3563
Riwayat versi ... 3564
Penyimpanan objek, penyimpanan file, dan caching file ... 3569
Amazon S3 ... 3569
Amazon EFS ... 3575
Amazon FSx ... 3578
Cache File Amazon ... 3584
Kelola sumber daya ... 3585
Pilih Wilayah untuk sumber daya Anda ... 3585
Temukan sumber daya Anda ... 3587
Langkah-langkah konsol ... 3587
Contoh baris perintah ... 3596
Tampilan Global (lintas Wilayah) ... 3600
AWS Tampilan Global ... 3600
Tandai sumber daya Anda ... 3604
Dasar-dasar tag ... 3605
Tandai sumber daya Anda ... 3607
Pembatasan tanda ... 3608
Manajemen tanda dan akses ... 3609
Menandai sumber daya Anda untuk penagihan ... 3609
Tandai izin sumber daya ... 3610
Menambahkan dan menghapus tag ... 3613
Filter sumber daya berdasarkan tag ... 3618
Melihat tag menggunakan metadata instance ... 3620
Kuota layanan ... 3626
Melihat kuota Anda saat ini ... 3626
Meminta peningkatan ... 3627
Pembatasan pada email yang dikirim menggunakan port 25 ... 3627

Pantau sumber daya ..... 3629
Memantau status instans Anda ..... 3630
Pemeriksaan status ..... 3631
Pemeriksaan status aplikasi ..... 3641
Peristiwa perubahan status ..... 3662
Peristiwa terjadwal ..... 3664
Memantau instans Anda menggunakan CloudWatch ..... 3703
Alarm instans ..... 3704
Kelola pemantauan terperinci ..... 3706
CloudWatch metrik ..... 3709
Instal dan konfigurasikan CloudWatch agen ..... 3729
Statistik untuk metrik ..... 3733
Lihat grafik pemantauan ..... 3743
Membuat alarm ..... 3744
Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans ..... 3745
Otomatisasi menggunakan EventBridge ..... 3758
Jenis EC2 acara Amazon ..... 3758
Log panggilan API menggunakan CloudTrail ..... 3759
Acara manajemen API Amazon EC2 di CloudTrail ..... 3761
Contoh acara API Amazon EC2 ..... 3761
Koneksi audit yang dibuat menggunakan EC2 Instance Connect ..... 3762
Memantau aplikasi.NET dan SQL Server ..... 3764
Melacak penggunaan Tingkat Gratis Anda ..... 3765
Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025 ..... 3765
Lacak penggunaan Tingkat Gratis untuk akun yang dibuat sebelum 15 Juli 2025 ..... 3767
Laporan penagihan dan penggunaan ..... 3769
Contoh ..... 3770
Instans bare metal ..... 3771
Host Khusus ..... 3771
Instans Khusus ..... 3771
Optimisasi EBS ..... 3772
Reservasi Kapasitas ..... 3772
Inventarisasi instans EC2 Anda ..... 3772
Lokasi geografis ..... 3774
Spesifikasi perangkat keras ..... 3775

Image ..... 3776
Alamat IP dan nama host DNS ..... 3778
Aturan-aturan grup keamanan ..... 3779
Data pengguna ..... 3781
Sumber daya terkait ..... 3782
Pemecahan Masalah ..... 3783
Masalah peluncuran instance ..... 3783
Nama perangkat tidak valid ..... 3784
Batas instans terlampaui ..... 3785
Kapasitas instans tidak cukup ..... 3785
Konfigurasi yang diminta saat ini tidak didukung. Periksa dokumentasi untuk konfigurasi yang didukung. ..... 3786
Instans langsung terhenti ..... 3786
Izin tidak cukup ..... 3788
Penggunaan CPU tinggi segera setelah Windows dimulai (hanya instans Windows) ..... 3789
Meluncurkan IMDSv1-enabled instance gagal ..... 3790
Masalah penghentian instans ..... 3791
Memaksa menghentikan instance ..... 3792
(Opsional) Buat instance pengganti ..... 3794
Masalah penghentian instans ..... 3796
Instans langsung terhenti ..... 3797
Penghentian instans yang tertunda ..... 3797
Instans yang dihentikan masih ditampilkan ..... 3799
Kesalahan: Instans mungkin tidak dihentikan. Ubah atribut instance 'non ApiTermination aktifkan' ..... 3799
Instans diluncurkan atau dihentikan secara otomatis ..... 3800
Instans yang tidak dapat dijangkau ..... 3800
Boot ulang instans ..... 3800
Output konsol instans ..... 3801
Mengambil tangkapan layar instans yang tidak dapat dijangkau ..... 3802
Screenshot umum untuk instance Windows ..... 3803
Pemulihan instans saat komputer host gagal ..... 3813
Instance muncul offline dan secara tak terduga di-reboot ..... 3813
Masalah SSH instance Linux ..... 3814
Penyebab umum masalah koneksi ..... 3815
Kesalahan saat menghubungkan instans Anda: Waktu koneksi habis ..... 3817

Kesalahan: tidak dapat memuat kunci ... Mengharapkan: KUNCI PRIVAT APA PUN ..... 3820
Kesalahan: Kunci pengguna tidak dikenali oleh server ..... 3821
Kesalahan: Izin ditolak atau koneksi ditutup oleh [instans] port 22 ..... 3823
Kesalahan: File kunci privat yang tidak dilindungi ..... 3825
Kesalahan: Kunci privat harus dimulai dengan “-----BEGIN RSA PRIVATE KEY-----” dan diakhiri dengan “-----END RSA PRIVATE KEY-----” ..... 3827
Kesalahan: Verifikasi kunci host gagal ..... 3827
Kesalahan: Server menolak kunci kami atau Tidak ada metode otentikasi yang didukung tersedia ..... 3828
Tidak dapat melakukan ping pada instans ..... 3829
Kesalahan: Server menutup koneksi jaringan secara tidak terduga ..... 3829
Kesalahan: Validasi kunci host gagal untuk EC2 Instance Connect ..... 3830
Tidak dapat terhubung ke instans Ubuntu menggunakan EC2 Instance Connect ..... 3832
Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance saya? ..... 3832
Pemeriksaan status instans Linux gagal ..... 3839
Meninjau informasi pemeriksaan status ..... 3840
Mengambil log sistem ..... 3841
Memecahkan masalah kesalahan log sistem untuk instans Linux ..... 3842
Kehabisan memori: hentikan proses ..... 3843
KESALAHAN: mmu_update gagal (Pembaruan manajemen memori gagal) ..... 3844
I/O kesalahan (kegagalan perangkat blok) ..... 3845
I/O KESALAHAN: bukan disk lokal maupun jarak jauh (Perangkat blok terdistribusi rusak) 3847
request_module: modprobe loop runaway (Melakukan loop modprobe kernel warisan pada versi Linux yang lebih lawas) ..... 3848
“FATAL: kernel terlalu lawas” dan “fsck: Tidak ada file atau direktori tersebut saat mencoba membuka /dev” (Kernel dan AMI tidak cocok) ..... 3849
“FATAL: Tidak dapat memuat/lib/modules" atau BusyBox "" (Modul kernel hilang) ..... 3850
KESALAHAN Kernel tidak valid (kernel tidak kompatibel dengan EC2) ..... 3852
fsck: Tidak ada file atau direktori tersebut saat mencoba membuka... (Sistem file tidak ditemukan) ..... 3853
Kesalahan umum saat memasang sistem file (kegagalan pemasangan) ..... 3855
VFS: Tidak dapat memasang fs root pada blok yang tidak diketahui (Sistem file root tidak cocok) ..... 3858
Kesalahan: Tidak dapat menentukan major/minor jumlah perangkat root... (Ketidak system/device cocokan file root) ..... 3859
XENBUS: Perangkat tanpa driver... ..... 3860

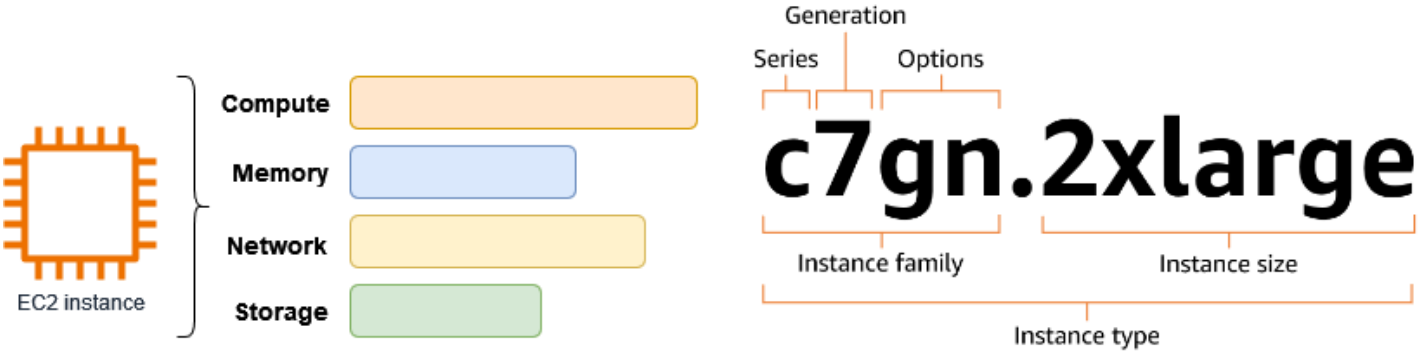
... hari tanpa diperiksa, pemeriksaan paksa (Diperlukan pemeriksaan sistem file) ..... 3862
fsck mati dengan status keluar... (Perangkat tidak ada) ..... 3862
Prompt GRUB (gru > bdom) ..... 3864
Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. (Alamat Hard-coded MAC) ..... 3867
Tidak dapat memuat Kebijakan SELinux. Mesin berada dalam mode pemberlakuan. Menghentikan sekarang. (Kesalahan konfigurasi SELinux) ..... 3868
XENBUS: Waktu menghubungkan ke perangkat habis (waktu habis Xenbus) ..... 3870
Instance Linux melakukan boot dari volume yang salah ..... 3871
Masalah RDP instance Windows ..... 3873
Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh ..... 3873
Kesalahan menggunakan klien RDP macOS ..... 3875
RDP menampilkan layar hitam, bukan desktop ..... 3875
Tidak dapat masuk dari jarak jauh ke sebuah instans dengan pengguna yang bukan administrator ..... 3876
Memecahkan masalah Remote Desktop menggunakan AWS Systems Manager ..... 3876
Aktifkan Desktop Jarak Jauh pada instans EC2 dengan registri jarak jauh ..... 3880
Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance Windows? ..... 3881
Masalah mulai instans Windows ..... 3882
“Kata sandi tidak tersedia” ..... 3882
“Kata sandi belum tersedia” ..... 3883
“Tidak dapat mengambil kata sandi Windows” ..... 3884
“Menunggu layanan metadata” ..... 3884
“Tidak dapat mengaktifkan Windows” ..... 3888
“Windows tidak asli (0x80070005)” ..... 3890
“Tidak ada Server Lisensi Server Terminal yang tersedia untuk memberikan lisensi” ..... 3891
“Beberapa pengaturan dikelola oleh organisasi Anda” ..... 3891
Masalah instance Windows ..... 3892
Tidak dapat terhubung AWS Systems Manager Manajer Sesi ke instans Windows Server 2025 ..... 3892
Volume EBS tidak diinisialisasi di Windows Server 2016 dan 2019 ..... 3893
Lakukan boot instans Windows EC2 ke Directory Service Restore Mode (DSRM) ..... 3894
Instans kehilangan konektivitas jaringan atau tugas terjadwal tidak berjalan saat diharapkan ..... 3897
Tidak bisa mendapatkan output konsol ..... 3898
Windows Server 2012 R2 tidak tersedia di jaringan ..... 3898

Tabrakan tanda tangan disk ... 3898
Windows instance kernel debug melalui jaringan ... 3900
Prasyarat ... 3900
Langkah 1: Instal Windows Debugging Tools pada host debug ... 3901
Langkah 2: Siapkan target debug ... 3902
Langkah 3: Mulai sesi debugging pada host debug ... 3904
Langkah 4: Reboot target debug ... 3905
Bersihkan pengaturan debug setelah debugging ... 3905
Batasan ... 3906
Catatan tambahan ... 3906
Setel ulang kata sandi administrator Windows ... 3907
Setel ulang kata sandi menggunakan EC2launch v2 ... 3908
Setel ulang kata sandi menggunakan EC2launch ... 3913
Setel ulang kata sandi menggunakan EC2config ... 3919
Memecahkan masalah Sysprep ... 3925
EC2Rescue untuk instans Linux ... 3927
Instal EC2Rescue ... 3927
Jalankan perintah EC2Rescue ... 3932
Mengembangkan modul EC2Rescue ... 3934
EC2Rescue untuk instans Windows ... 3942
Memecahkan masalah menggunakan EC2Rescue GUI ... 3943
Memecahkan masalah menggunakan EC2Rescue CLI ... 3949
Memecahkan masalah menggunakan EC2Rescue dan Systems Manager ... 3958
Konsol Serial EC2 ... 3962
Prasyarat ... 3962
Konfigurasikan akses ke Konsol Serial EC2 ... 3971
Hubungkan ke Konsol Serial EC2 ... 3982
Memutuskan koneksi dari Konsol Serial EC2 ... 3992
Memecahkan masalah instans Anda menggunakan Konsol Serial EC2 ... 3993
Kirim interupsi diagnostik ... 4001
Tipe instans yang didukung ... 4002
Prasyarat ... 4003
Kirimkan interupsi diagnostik ... 4005
Riwayat dokumen ... 4007
Sejarah untuk 2018 dan sebelumnya ... 4053
... mmmmlxxx

# Apa itu Amazon EC2?

Amazon Elastic Compute Cloud (Amazon EC2) menyediakan kapasitas komputasi yang sesuai permintaan dan dapat diskalakan di Amazon Web Services (AWS) Cloud. Menggunakan Amazon EC2 akan mengurangi biaya perangkat keras sehingga Anda dapat mengembangkan dan melakukan deployment aplikasi lebih cepat. Anda dapat menggunakan Amazon EC2 untuk meluncurkan server virtual sebanyak atau sesedikit yang Anda butuhkan, mengonfigurasi keamanan dan jaringan, serta mengelola penyimpanan. Anda dapat menambahkan kapasitas (menaikkan skala) untuk menangani tugas-tugas berat komputasi, seperti proses bulanan atau tahunan, atau lonjakan lalu lintas situs web. Ketika penggunaan berkurang, Anda dapat mengurangi kapasitas (menurunkan skala) lagi.

Instans EC2 adalah server virtual di AWS Cloud. Saat meluncurkan instans EC2, jenis instans yang Anda tentukan menentukan perangkat keras yang tersedia untuk instans Anda. Setiap jenis instans menawarkan keseimbangan sumber daya komputasi, memori, jaringan, dan penyimpanan yang berbeda. Untuk informasi selengkapnya, lihat Panduan Jenis Instans Amazon EC2.




# Fitur-fitur Amazon EC2

Amazon EC2 menyediakan fitur-fitur tingkat tinggi berikut ini:

Instans

Server virtual.

Amazon Machine Images (AMI)

Templat yang telah dikonfigurasi untuk instans Anda yang mengemas komponen yang Anda butuhkan untuk server Anda (termasuk sistem operasi dan perangkat lunak tambahan).

Tipe instans

Berbagai konfigurasi CPU, memori, penyimpanan, kapasitas jaringan, dan perangkat keras grafis untuk instans Anda.

Volume Amazon EBS

Volume penyimpanan persisten untuk data Anda menggunakan Amazon Elastic Block Store (Amazon EBS).

Volume penyimpanan instans

Volume penyimpanan untuk data sementara yang dihapus saat Anda menghentikan, hibernasi, atau mengakhiri instans Anda.

Pasangan kunci

Amankan informasi login untuk instans Anda. AWS menyimpan kunci publik dan Anda menyimpan kunci pribadi di tempat yang aman.

Grup keamanan

Firewall virtual yang memungkinkan Anda menentukan protokol, port, dan rentang IP sumber yang dapat menjangkau instans Anda, serta rentang IP tujuan yang dapat terhubung ke instans Anda.

Amazon EC2 mendukung pemrosesan, penyimpanan, dan transmisi data kartu kredit oleh pedagang atau penyedia layanan, serta telah divalidasi sesuai dengan Standar Keamanan Data (DSS) Industri Kartu Pembayaran (PCI). Untuk informasi selengkapnya tentang PCI DSS, termasuk cara meminta salinan PCI AWS Compliance Package, lihat PCI DSS Level 1.

# Layanan terkait

Layanan untuk digunakan dengan Amazon EC2

Anda dapat menggunakan instans lain Layanan AWS dengan instans yang Anda gunakan menggunakan Amazon EC2.

Amazon EC2 Auto Scaling

Membantu memastikan Anda memiliki jumlah instans Amazon EC2 yang tepat serta tersedia untuk menangani beban aplikasi Anda.

AWS Backup

Otomatiskan pencadangan instans Amazon EC2 Anda dan volume Amazon EBS yang dilampirkan padanya.

Amazon CloudWatch

Pantau instans dan volume Amazon EBS Anda.

Elastic Load Balancing

Mendistribusikan lalu lintas aplikasi yang masuk ke banyak instans secara otomatis.

Amazon GuardDuty

Deteksi penggunaan yang berpotensi tidak sah atau berbahaya dari instans EC2 Anda.

EC2 Image Builder

Mengotomatiskan pembuatan, pengelolaan, dan deployment gambar server yang disesuaikan, aman, dan terkini.

AWS Launch Wizard

Mengukur, mengonfigurasi, dan menyebarkan AWS sumber daya untuk aplikasi pihak ketiga tanpa harus mengidentifikasi dan menyediakan AWS sumber daya individual secara manual.

AWS Systems Manager

Lakukan operasi dalam skala besar pada instans EC2 dengan solusi manajemen end-to-end yang aman ini.

### Layanan komputasi tambahan

Anda dapat meluncurkan instans menggunakan layanan AWS komputasi lain alih-alih menggunakan Amazon EC2.

Amazon Lightsail

Buat situs web atau aplikasi web menggunakan Amazon Lightsail, platform cloud yang menyediakan sumber daya yang Anda butuhkan untuk menyebarkan proyek Anda dengan cepat, dengan harga bulanan yang rendah dan dapat diprediksi. Untuk membandingkan Amazon EC2 dan Lightsail, lihat Amazon Lightsail atau Amazon EC2.

Amazon Elastic Container Service (Amazon ECS)

Melakukan deployment, mengelola, dan menskalakan aplikasi dalam kontainer pada klaster instans EC2. Untuk informasi selengkapnya, lihat Memilih layanan AWS kontainer.

Amazon Elastic Kubernetes Service (Amazon EKS)

Jalankan aplikasi Kubernetes Anda di AWS. Untuk informasi selengkapnya, lihat Memilih layanan AWS kontainer.

# Akses Amazon EC2

Anda dapat membuat dan mengelola instans Amazon EC2 menggunakan antarmuka berikut:

Konsol Amazon EC2

Antarmuka web sederhana untuk membuat serta mengelola instans Amazon EC2 dan sumber daya. Jika Anda telah mendaftar untuk sebuah AWS akun, Anda dapat mengakses konsol Amazon EC2 dengan masuk ke Konsol Manajemen AWS dan memilih EC2 dari halaman beranda konsol.

AWS Command Line Interface

Memungkinkan Anda berinteraksi dengan AWS layanan menggunakan perintah di shell baris perintah Anda. Hal ini didukung di Windows, Mac, dan Linux. Untuk informasi tentang AWS CLI selengkapnya, lihat Panduan Pengguna AWS Command Line Interface. Anda dapat menemukan perintah Amazon EC2 di Referensi Perintah AWS CLI.

CloudFormation

Amazon EC2 mendukung pembuatan sumber daya menggunakan CloudFormation. Anda membuat template, dalam format JSON atau YAMAL, yang menjelaskan AWS sumber daya Anda, dan CloudFormation ketentuan serta mengonfigurasi sumber daya tersebut untuk Anda. Anda dapat menggunakan kembali CloudFormation template untuk menyediakan resource yang sama beberapa kali, baik di Region dan akun yang sama atau di beberapa Region dan akun. Untuk informasi tentang tipe sumber daya yang didukung dan properti pada Amazon EC2 selengkapnya, lihat referensi tipe sumber daya EC2 di Panduan Pengguna AWS CloudFormation .

AWS SDK

Jika Anda lebih suka membangun aplikasi menggunakan API khusus bahasa daripada mengirimkan permintaan melalui HTTP atau HTTPS, AWS menyediakan pustaka, kode sampel, tutorial, dan sumber daya lainnya untuk pengembang perangkat lunak. Pustaka ini menyediakan fungsi dasar yang mengotomatiskan tugas-tugas seperti menandatangani permintaan Anda secara kriptografis, mencoba kembali permintaan, dan menangani respons kesalahan, sehingga memudahkan Anda untuk memulai. Untuk informasi selengkapnya, lihat Alat untuk Membangun di AWS.

Alat AWS untuk PowerShell

Satu set PowerShell modul yang dibangun di atas fungsionalitas yang diekspos oleh SDK untuk .NET. Alat untuk PowerShell memungkinkan Anda melakukan operasi skrip pada AWS sumber daya Anda dari baris PowerShell perintah. Untuk mulai, lihat Panduan Pengguna Alat AWS untuk PowerShell. Anda dapat menemukan cmdlet untuk Amazon EC2, di Referensi Cmdlet Alat AWS untuk PowerShell.

API Kueri

Amazon EC2 menyediakan API Kueri. Permintaan ini adalah permintaan HTTP atau HTTPS yang menggunakan parameter HTTP verbs GET atau POST dan parameter Kueri yang diberi nama `Action`. Untuk informasi tentang tindakan API untuk Amazon EC2 selengkapnya, lihat Tindakan di Referensi API Amazon EC2.

# Harga untuk Amazon EC2

Amazon EC2 menyediakan opsi harga berikut:

Tingkat Gratis

Anda dapat memulai Amazon EC2 secara gratis. Untuk menjelajahi opsi Tingkat Gratis, lihat AWS Tingkat Gratis.

On-Demand Contoh

Bayar instans yang Anda gunakan dalam hitungan detik, dengan minimal 60 detik, tanpa komitmen jangka panjang atau pembayaran di muka.

Savings Plans

Anda dapat mengurangi biaya Amazon EC2 dengan membuat komitmen pada jumlah penggunaan yang konsisten, dalam USD per jam, untuk jangka waktu 1 atau 3 tahun.

Instans Terpesan

Anda dapat mengurangi biaya Amazon EC2 dengan membuat komitmen pada konfigurasi instans tertentu, termasuk tipe dan Wilayah instans, untuk jangka waktu 1 atau 3 tahun.

Instans Spot

Minta instans EC2 yang tidak digunakan, yang dapat mengurangi biaya Amazon EC2 Anda secara signifikan.

Host Khusus

Mengurangi biaya dengan menggunakan server EC2 fisik yang sepenuhnya didedikasikan untuk Anda gunakan, baik On-Demand atau sebagai bagian dari Savings Plan. Anda dapat menggunakan lisensi perangkat lunak terikat server yang ada dan mendapatkan bantuan untuk memenuhi persyaratan kepatuhan.

On-Demand Reservasi Kapasitas

Cadangkan kapasitas komputasi untuk instans EC2 Anda di Zona Ketersediaan tertentu untuk durasi waktu apa pun.

Per-second penagihan

Menghapus biaya menit dan detik yang tidak terpakai dari tagihan Anda.

Untuk daftar lengkap biaya dan harga Amazon EC2 serta informasi tentang model pembelian selengkapnya, lihat harga Amazon EC2.

## Estimasi, penagihan, dan optimalisasi biaya

Untuk membuat perkiraan untuk kasus AWS penggunaan Anda, gunakan AWS Kalkulator Harga.

Untuk memperkirakan biaya transformasi beban kerja Microsoft menjadi arsitektur modern yang menggunakan layanan open source dan cloud-native yang digunakan AWS, gunakan Kalkulator Modernisasi AWS untuk Beban Kerja Microsoft.

Untuk melihat tagihan Anda, buka Dasbor Manajemen Penagihan dan Biaya di konsol AWS Manajemen Penagihan dan Biaya. Tagihan Anda berisi tautan ke laporan penggunaan yang

memberikan detail tentang tagihan Anda. Untuk mempelajari lebih lanjut tentang penagihan AWS akun, lihat Panduan Pengguna AWS Billing and Cost Management.

Jika Anda memiliki pertanyaan tentang AWS penagihan, akun, dan acara, hubungi AWS Support.

Untuk menghitung biaya sampel lingkungan yang disediakan, lihat Pusat Ekonomi Cloud. Saat menghitung biaya lingkungan yang disediakan, ingatlah untuk menyertakan biaya insidental seperti penyimpanan snapshot untuk volume EBS.

Anda dapat mengoptimalkan biaya, keamanan, dan kinerja AWS lingkungan Anda menggunakan AWS Trusted Advisor.

Anda dapat menggunakan AWS Cost Explorer untuk menganalisis biaya dan penggunaan instans EC2 Anda. Anda dapat melihat data hingga 13 bulan terakhir, dan memperkirakan berapa banyak kemungkinan Anda akan menghabiskan untuk 12 bulan ke depan. Untuk informasi selengkapnya, lihat Menganalisis biaya dan penggunaan Anda AWS Cost Explorer di Panduan AWS Cost Management Pengguna.

## Sumber daya

- Fitur Amazon EC2
- AWS Re: posting
- AWS Pembangun Keterampilan
- AWS Support
- Hands-on Tutorial
- Web Hosting
- Windows aktif AWS

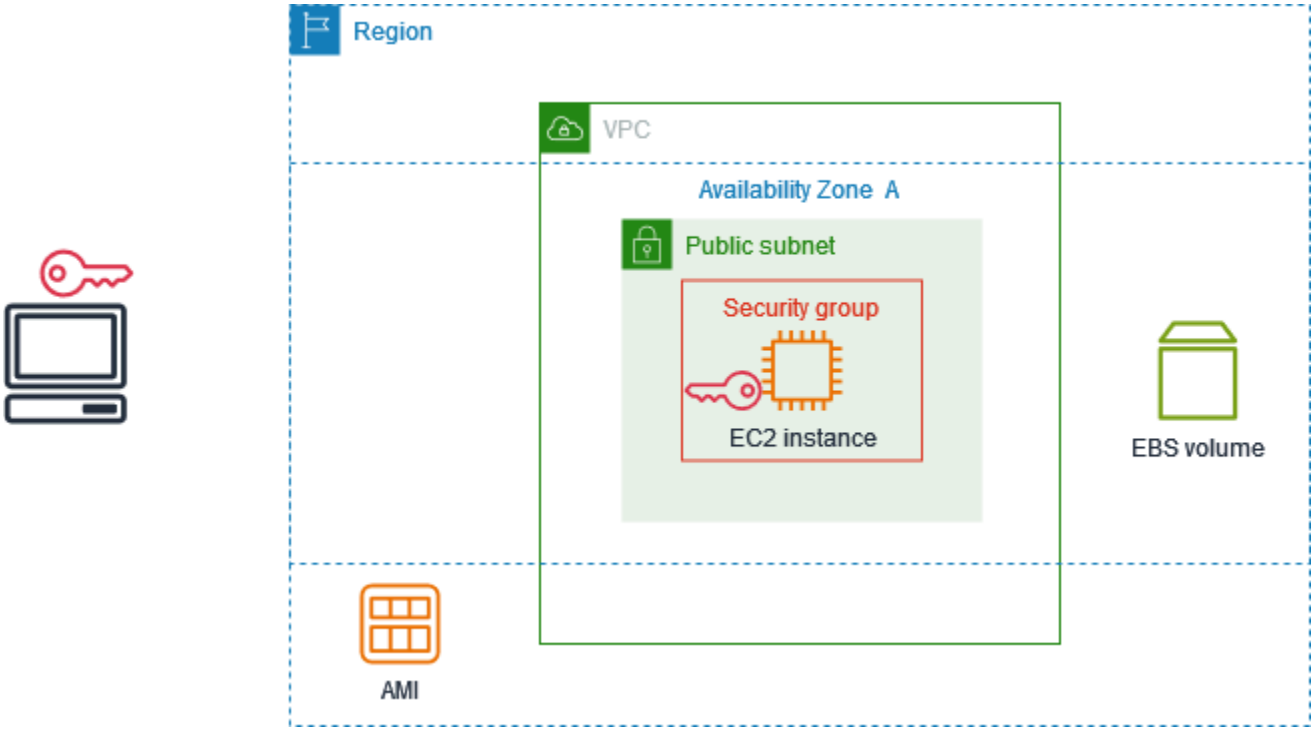
# Mulai Amazon EC2

Gunakan tutorial ini untuk memulai Amazon Elastic Compute Cloud (Amazon EC2). Anda akan belajar cara meluncurkan dan terhubung ke instance EC2. Sebuah  instance  adalah server virtual di AWS Cloud. Dengan Amazon EC2, Anda dapat menyiapkan serta mengonfigurasi sistem operasi dan aplikasi yang berjalan di instans.

Gambaran umum

Diagram berikut menunjukkan komponen kunci yang akan Anda gunakan dalam tutorial ini:

- Gambar  — Template yang berisi perangkat lunak untuk dijalankan pada instance Anda, seperti sistem operasi.
- Pasangan kunci  — Satu set kredentif keamanan yang Anda gunakan untuk membuktikan identitas Anda saat menghubungkan ke instans Anda. Kunci publik ada di instance Anda dan kunci pribadi ada di komputer Anda.
- Sebuah jaringan  — Virtual Private Cloud (VPC) adalah jaringan virtual yang didedikasikan untuk Anda Akun AWS. Untuk membantu Anda memulai dengan cepat, akun Anda dilengkapi dengan VPC default di masing-masing Wilayah AWS, dan setiap VPC default memiliki subnet default di setiap Zona Ketersediaan.
- Grup keamanan  — Bertindak sebagai firewall virtual untuk mengontrol lalu lintas masuk dan keluar.
- Volume EBS  — Volume root diperlukan untuk gambar. Anda dapat menambahkan volume data secara opsional.




Biaya untuk tutorial ini

Saat Anda membuat Akun AWS, Anda dapat memulai dengan Amazon EC2 secara gratis menggunakan. AWS Tingkat Gratis

Jika Anda membuat Akun AWS sebelum 15 Juli 2025, usianya kurang dari 12 bulan, dan Anda belum melebihi manfaat Tingkat Gratis untuk Amazon EC2, Anda tidak akan dikenakan biaya apa pun untuk menyelesaikan tutorial ini, karena tutorial ini membantu Anda memilih opsi yang ada dalam manfaat Tingkat Gratis. Jika tidak, Anda akan dikenakan biaya penggunaan Amazon EC2 standar dari saat Anda meluncurkan instans (meskipun instans tetap idle) hingga Anda menghentikannya.

Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, usianya kurang dari 6 bulan, dan Anda belum menghabiskan semua kredit Anda, tidak akan dikenakan biaya apa pun untuk menyelesaikan tutorial ini, karena tutorial ini membantu Anda memilih opsi yang ada dalam manfaat Tingkat Gratis.

Untuk informasi tentang cara menentukan apakah Anda memenuhi syarat untuk Tingkat Gratis, lihatthe section called “Melacak penggunaan Tingkat Gratis Anda”.

Tugas

- Langkah 1: Luncurkan instans
- Langkah 2: Hubungkan ke instans Anda

- Langkah 3: Bersihkan instans Anda
- Langkah selanjutnya


# Langkah 1: Luncurkan instans

Anda dapat meluncurkan instance EC2 menggunakan Konsol Manajemen AWS seperti yang dijelaskan dalam prosedur berikut. Tutorial ini dimaksudkan untuk membantu Anda meluncurkan instans pertama dengan cepat, jadi tutorial ini tidak mencakup semua opsi yang memungkinkan.

Untuk meluncurkan sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di bilah navigasi di bagian atas layar, arus Wilayah AWS ditampilkan - misalnya, Ohio. Anda dapat menggunakan Wilayah yang dipilih, atau secara opsional memilih Wilayah yang lebih dekat dengan Anda.
3. Dari dasbor konsol EC2, di panel Luncurkan instance, pilih Luncurkan instance.
4. Di bawah Nama dan tanda, untuk Nama, masukkan nama deskriptif untuk instans Anda.
5. Di bawah Aplikasi dan Citra OS (Amazon Machine Image), lakukan hal berikut:
   a. Pilih Mulai C epat, lalu pilih sistem operasi (OS) untuk instance Anda. Untuk contoh Linux pertama Anda, kami sarankan Anda memilih Amazon Linux.
   b. Dari Amazon Machine Image (AMI), pilih AMI yang bertanda Lay ak Tingkat Gratis .
6. Di bawah Jenis Instans, untuk Jenis Instans, pilih jenis instans yang ditandai Layak Tingkat Gratis .
7. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang ada atau pilih Buat pasangan kunci baru untuk membuat pasangan kunci pertama Anda.

   > ⚠ Warning
   >
   > Jika Anda memilih Lanjutkan tanpa pasangan kunci (Tidak disarankan), Anda tidak akan dapat terhubung ke instance Anda menggunakan metode yang dijelaskan dalam tutorial ini.

8. Di bawah Pengaturan jaringan, perhatikan bahwa VPC default Anda dipilih, opsi untuk menggunakan subnet default di Zona Ketersediaan yang dipilih untuk Anda, dan grup keamanan dengan aturan yang memungkinkan koneksi ke instance Anda dari mana saja ()`0.0.0.0/0`.

> **Warning**
>
> Jika Anda menentukan `0.0.0.0/0`, Anda mengaktifkan lalu lintas dari alamat IP manapun di dunia. Untuk protokol SSH dan RDP, Anda mungkin menganggap ini dapat diterima untuk waktu yang singkat di lingkungan pengujian, tetapi tidak aman untuk lingkungan produksi. Dalam produksi, pastikan untuk mengotorisasi akses hanya dari alamat IP individu atau rentang alamat yang sesuai.

Untuk contoh pertama Anda, kami sarankan Anda menggunakan pengaturan default. Jika tidak, Anda dapat memperbarui pengaturan jaringan Anda sebagai berikut:

- (Opsional) Untuk menggunakan subnet default tertentu, pilih Edit lalu pilih subnet.
- (Opsional) Untuk menggunakan VPC yang berbeda, pilih Edit lalu pilih VPC yang ada. Jika VPC tidak dikonfigurasi untuk akses internet publik, Anda tidak akan dapat terhubung ke instans.
- (Opsional) Untuk membatasi lalu lintas koneksi masuk ke jaringan tertentu, pilih K ustom al ih-alih Di mana saja, dan masukkan blok CIDR untuk jaringan Anda.
- (Opsional) Untuk menggunakan grup keamanan yang berbeda, pilih Pilih grup keamanan yang ada dan pilih grup keamanan yang ada. Jika grup keamanan tidak memiliki aturan yang mengizinkan lalu lintas koneksi dari jaringan Anda, Anda tidak akan dapat terhubung ke instans. Untuk instance Linux, Anda harus mengizinkan lalu lintas SSH. Untuk contoh Windows, Anda harus mengizinkan lalu lintas RDP.

9. Di bawah Konfigurasi penyimpanan, perhatikan bahwa volume root dikonfigurasi tetapi tidak ada volume data. Ini cukup untuk tujuan pengujian.
10. Tinjau ringkasan konfigurasi instans di panel Ringkasan, dan ketika Anda siap, pilih Luncurkan instans.
11. Jika peluncuran berhasil, pilih ID instance dari notifikasi Sukses untuk membuka halaman Instans dan memantau status peluncuran.
12. Pilih kotak centang untuk instans. Status contoh awal adalah`pending`. Setelah instance dimulai, statusnya berubah menjadi`running`. Pilih tab Status dan alarm. Setelah instans Anda melewati pemeriksaan statusnya, instans siap menerima permintaan koneksi.

# Langkah 2: Hubungkan ke instans Anda

Prosedur yang Anda gunakan tergantung pada sistem operasi instance. Jika Anda tidak dapat terhubung ke instans, lihat Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux untuk mendapatkan bantuan.

## Instans Linux

Anda dapat terhubung ke instance Linux Anda menggunakan klien SSH apa pun. Jika Anda menjalankan Windows di komputer Anda, buka terminal dan jalankan `ssh` perintah untuk memverifikasi bahwa Anda telah menginstal klien SSH. Jika perintah tidak ditemukan, instal OpenSSH untuk Windows.

Untuk terhubung ke instans Anda menggunakan SSH

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Hubungkan.
4. Pada halaman Connect to instance, pilih tab klien SSH.
5. (Opsional) Jika Anda membuat pasangan kunci saat meluncurkan instance dan mengunduh kunci pribadi (file.pem) ke komputer yang menjalankan Linux atau macOS, jalankan chmod perintah contoh untuk mengatur izin untuk kunci pribadi Anda.
6. Salin contoh perintah SSH. Berikut ini adalah contoh, di mana *key-pair-name* .pem adalah nama file kunci pribadi Anda, *ec2-user* adalah nama pengguna yang terkait dengan gambar, dan string setelah simbol @ adalah nama DNS publik dari instance.

   ```
   ssh -i key-pair-name.pem ec2-user@ec2-198-51-100-1.us-east-2.compute.amazonaws.com
   ```

7. Di jendela terminal di komputer Anda, jalankan ssh perintah yang Anda simpan di langkah sebelumnya. Jika file kunci pribadi tidak ada di direktori saat ini, Anda harus menentukan jalur yang sepenuhnya memenuhi syarat ke file kunci dalam perintah ini.

   Berikut adalah respons contohnya:

   ```
   The authenticity of host 'ec2-198-51-100-1.us-east-2.compute.amazonaws.com
    (198-51-100-1)' can't be established.
   ECDSA key fingerprint is l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
   Are you sure you want to continue connecting (yes/no)?
   ```

8. (Opsional) Pastikan sidik jari dalam peringatan keamanan cocok dengan sidik jari instans yang terdapat dalam output konsol saat Anda memulai instance pertama kali. Untuk mendapatkan output konsol, pilih Tindakan , Monitor dan pemecahan masalah, Dapatkan log sistem. Jika sidik jari tidak cocok, seseorang mungkin mencoba melakukan serangan man-in-the-middle. Jika cocok, lanjutkan ke langkah berikutnya.
9. Masukkan **`yes`**.

   Berikut adalah respons contohnya:

```
Warning: Permanently added 'ec2-198-51-100-1.us-
east-2.compute.amazonaws.com' (ECDSA) to the list of known hosts.
```

## Instans Windows

Untuk menyambung ke instance Windows menggunakan RDP, Anda harus mengambil kata sandi administrator awal dan kemudian memasukkan kata sandi ini saat Anda terhubung ke instance Anda. Diperlukan beberapa menit setelah peluncuran instans sebelum sandi ini tersedia. Akun Anda harus memiliki izin untuk melakukan GetPasswordData tindakan. Untuk informasi selengkapnya, lihat Contoh kebijakan untuk mengontrol akses Amazon EC2 API.

Nama pengguna default untuk akun Administrator tergantung pada bahasa sistem operasi (OS) yang terdapat dalam AMI. Untuk menentukan nama pengguna yang benar, identifikasi bahasa OS, lalu pilih nama pengguna yang sesuai. Misalnya, untuk OS Inggris, nama penggunanya adalah`Administrator`, untuk OS Prancis itu`Administrateur`, dan untuk OS Portugis itu`Administrador`. Jika versi bahasa OS tidak memiliki nama pengguna dalam bahasa yang sama, pilih nama penggguna`Administrator (Other)`. Untuk informasi selengkapnya, lihat Nama Lokalisasi untuk Akun Administrator di Windows di situs web Microsoft.

Untuk mengambil kata sandi administrator awal

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Hubungkan.
4. Pada halaman Connect to instance, pilih tab klien RDP.
5. Untuk Nama Pengguna, pilih nama pengguna default untuk akun Administrator. Nama pengguna yang Anda pilih harus sesuai dengan bahasa sistem operasi (OS) yang terdapat

dalam AMI yang Anda gunakan untuk meluncurkan instans Anda. Jika tidak ada nama pengguna dalam bahasa yang sama dengan OS Anda, pilih Administrator (Lainnya).

6. Pilih Dapatkan kata sandi.
7. Pada halaman Dapatkan kata sandi Windows, lakukan hal berikut:

   a. Pilih Unggah file kunci pribadi dan arahkan ke file private key (`.pem`) yang Anda tentukan saat meluncurkan instance. Pilih file dan pilih Buka untuk menyalin seluruh isi file ke jendela ini.

   b. Pilih Dek ripsi kata sandi. Halaman Dapatkan kata sandi Windows ditutup, dan kata sandi administrator default untuk instance muncul di bawah Kata Sand i, menggantikan tautan Dapatkan kata sandi yang ditampilkan sebelumnya.

   c. Salin kata sandi dan simpan di tempat yang aman. Kata sandi ini diperlukan untuk terhubung ke instans.

Prosedur berikut menggunakan klien Remote Desktop Connection untuk Windows (MSTSC). Jika Anda menggunakan klien RDP yang berbeda, unduh file RDP dan kemudian lihat dokumentasi untuk klien RDP untuk langkah-langkah untuk membuat koneksi RDP.

Untuk menyambung ke instance Windows menggunakan klien RDP

1. Pada halaman Hubungkan ke instance, pilih Unduh file desktop jarak jauh. Setelah pengunduhan file selesai, pilih Bat al untuk kembali ke halaman Instans. File RDP diunduh ke `Downloads` folder Anda.
2. Jalan `mstsc.exe` kan untuk membuka klien RDP.
3. Perluas Tampilkan opsi, pilih Buka, dan pilih file.rdp dari folder Anda`Downloads`.
4. Secara default, Komputer adalah nama DNS IPv4 publik dari instance dan Nama Pengguna adalah akun administrator. Untuk menyambung ke instance menggunakan IPv6 sebagai gantinya, ganti nama DNS IPv4 publik dari instance dengan alamat IPv6-nya. Tinjau pengaturan default dan ubah sesuai kebutuhan.
5. Pilih Hubungkan. Jika Anda menerima peringatan bahwa penerbit koneksi jarak jauh tidak diketahui, pilih Hubun gkan untuk melanjutkan.
6. Masukkan kata sandi yang Anda simpan sebelumnya, lalu pilih OK.
7. Karena sifat dari sertifikat yang ditandatangani sendiri, Anda mungkin mendapatkan peringatan bahwa sertifikat keamanan tidak dapat diautentikasi. Lakukan salah satu tindakan berikut:
   - Jika Anda mempercayai sertifikat, pilih Ya untuk menyambung ke instans Anda.

- [Windows] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai di log sistem untuk mengkonfirmasi identitas komputer jarak jauh. Pilih Lihat sertifikat dan kemudian pilih Thumbprint dari tab Detail. Bandingkan nilai ini dengan nilai `RDPCERTIFICATE-THUMBPRINT` di Tindakan , Monitor dan pemecahan masalah, Dapatkan log sistem.
- [Mac OS X] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai di log sistem untuk mengonfirmasi identitas komputer jarak jauh. Pilih Tampilkan Sertifikat, perluas Detail, dan pilih Sidik J ari SHA1. Bandingkan nilai ini dengan nilai `RDPCERTIFICATE-THUMBPRINT` di Tindakan , Monitor dan pemecahan masalah, Dapatkan log sistem.

8. Jika koneksi RDP berhasil, klien RDP menampilkan layar login Windows dan kemudian desktop Windows. Jika Anda menerima pesan kesalahan sebagai gantinya, lihatthe section called "Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh". Ketika Anda selesai dengan koneksi RDP, Anda dapat menutup klien RDP.

## Langkah 3: Bersihkan instans Anda

> **Warning**
>
> Mengakhiri instance bersifat permanen dan tidak dapat diubah.
> Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
> Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Setelah selesai dengan instans yang Anda buat untuk tutorial ini, Anda harus membersihkannya dengan mengakhiri instans tersebut. Jika Anda ingin melakukan lebih banyak hal dengan instans ini sebelum membersihkannya, lihat Langkah selanjutnya.

Anda akan berhenti dikenakan biaya untuk instans atau penggunaan yang diperhitungkan terhadap batas Tingkat Gratis Anda segera setelah status instans berubah menjadi `shutting down` atau`terminated`. Untuk menyimpan instans untuk nanti, tetapi tidak dikenakan biaya atau

penggunaan yang diperhitungkan terhadap batas Tingkat Gratis Anda, Anda dapat menghentikan instans sekarang dan kemudian memulainya lagi nanti. Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.

Untuk mengakhiri instans Anda

1. Di panel navigasi, pilih Instans. Dalam daftar instans, pilih instans tersebut.
2. Pilih Status Instance, Hentikan (hapus) instance.
3. Pilih Hentikan (hapus) saat diminta konfirmasi.

   Amazon EC2 akan tertutup dan mengakhiri instans Anda. Setelah Anda mengakhiri sebuah instans, instans akan tetap terlihat di konsol untuk beberapa saat, kemudian entri tersebut akan dihapus secara otomatis. Anda tidak dapat menghapus sendiri instans yang telah diakhiri dari tampilan konsol.

## Langkah selanjutnya

Setelah memulai instance, Anda mungkin ingin menjelajahi langkah-langkah berikut:

- Jelajahi konsep inti Amazon EC2 dengan tutorial pengantar. Untuk informasi selengkapnya, lihat Tutorial untuk meluncurkan instans EC2.
- Pelajari cara melacak penggunaan Tingkat Gratis Amazon EC2 menggunakan konsol. Untuk informasi selengkapnya, lihat the section called “Melacak penggunaan Tingkat Gratis Anda”.
- Konfigurasikan CloudWatch alarm untuk memberi tahu Anda jika penggunaan Anda melebihi Tingkat Gratis (untuk akun yang dibuat sebelum 15 Juli 2025). Untuk informasi selengkapnya, lihat Mel AWS Tingkat Gratis acak penggunaan Anda di Panduan AWS Billing Pengguna.
- Tambahkan volume EBS. Untuk informasi selengkapnya, lihat Membuat volume Amazon EBS di Panduan Pengguna Amazon EBS.
- Pelajari cara mengelola instans EC2 Anda dari jarak jauh menggunakan perintah Run. Untuk informasi selengkapnya, lihat Run Command AWS Systems Manager di Panduan Pengguna AWS Systems Manager .
- Pelajari tentang opsi pembelian instans. Untuk informasi selengkapnya, lihat Opsi penagihan dan pembelian Amazon EC2.
- Dapatkan saran tentang tipe instans. Lihat informasi yang lebih lengkap di Dapatkan rekomendasi dari pencari tipe instans EC2.

# Praktik terbaik untuk Amazon EC2

Untuk mendapatkan keuntungan maksimal dari Amazon EC2, kami menyarankan Anda untuk melakukan praktik terbaik berikut.

Keamanan

- Kelola akses ke AWS sumber daya dan APIs gunakan federasi identitas dengan penyedia identitas dan peran IAM bila memungkinkan. Untuk informasi selengkapnya, lihat Penyedia dan federasi identitas dalam Panduan Pengguna IAM.
- Terapkan aturan paling sedikit permisif untuk grup keamanan Anda.
- Lakukan patch, perbarui, serta amankan sistem operasi dan aplikasi pada instans Anda secara berkala. Untuk informasi selengkapnya, lihat Manajemen pembaruan. Untuk panduan khusus untuk sistem operasi Windows, lihatPraktik terbaik keamanan untuk instans Windows.
- Gunakan Amazon Inspector untuk secara otomatis menemukan dan memindai instans Amazon EC2 agar dapat memeriksa kerentanan perangkat lunak dan paparan jaringan yang tidak diinginkan. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon Inspector.
- Gunakan AWS Security Hub CSPM kontrol untuk memantau sumber daya Amazon EC2 Anda terhadap praktik terbaik keamanan dan standar keamanan. Untuk informasi selengkapnya tentang menggunakan Security Hub CSPM, lihat kontrol Amazon Elastic Compute Cloud di Panduan Pengguna.AWS Security Hub CSPM

Penyimpanan

- Memahami implikasi dari tipe volume root untuk persistensi data, backup, dan recovery. Untuk informasi selengkapnya, lihat Tipe volume root.
- Gunakan volume Amazon EBS yang terpisah untuk sistem operasi versus data Anda. Pastikan bahwa volume dengan data Anda tetap ada setelah instans diakhiri. Untuk informasi selengkapnya, lihat Pertahankan data saat instans diakhiri.
- Gunakan penyimpanan instans yang tersedia untuk instans Anda agar dapat menyimpan data sementara. Perlu diingat bahwa data yang disimpan di penyimpanan instans akan dihapus saat Anda menghentikan, tidak mengaktifkan sementara, atau mengakhiri instans. Jika Anda menggunakan penyimpanan instans untuk penyimpanan basis data, pastikan bahwa Anda memiliki klaster dengan faktor replikasi yang menjamin toleransi kesalahan.

- Enkripsikan volume dan snapshot EBS. Untuk informasi selengkapnya, lihat enkripsi Amazon EBS di Panduan Pengguna Amazon EBS.

Manajemen sumber daya

- Gunakan tanda metadata instans dan sumber daya kustom untuk melacak dan mengidentifikasi sumber daya AWS Anda. Untuk informasi lebih lanjut, lihat Menggunakan metadata instans untuk mengelola instans EC2 Anda dan Tandai sumber daya Amazon EC2 Anda.
- Lihat batas Anda saat ini untuk Amazon EC2. Rencanakan untuk meminta kenaikan batas sebelum Anda membutuhkannya. Untuk informasi selengkapnya, lihat Kuota layanan Amazon EC2.
- Gunakan AWS Trusted Advisor untuk memeriksa AWS lingkungan Anda, dan kemudian membuat rekomendasi ketika ada peluang untuk menghemat uang, meningkatkan ketersediaan dan kinerja sistem, atau membantu menutup kesenjangan keamanan. Untuk informasi selengkapnya, lihat AWS Trusted Advisor dalam Panduan Pengguna AWS Dukungan .

Pencadangan dan pemulihan

- Cadangkan volume EBS secara berkala menggunakan snapshot Amazon EBS, dan buat Amazon Machine Image (AMI) dari instans Anda agar dapat menyimpan konfigurasi sebagai templat untuk meluncurkan instans mendatang. Untuk informasi selengkapnya tentang AWS layanan yang membantu mencapai kasus penggunaan ini, lihat AWS Backupdan Amazon Data Lifecycle Manager.
- Deploy komponen penting aplikasi Anda di banyak Zona Ketersediaan, dan replikasi data Anda dengan tepat.
- Desain aplikasi Anda untuk menangani pembuatan alamat IP dinamis saat instans Anda dimulai ulang. Untuk informasi selengkapnya, lihat Pengalamatan IP instans Amazon EC2.
- Pantau dan respons peristiwa. Untuk informasi selengkapnya, lihat Pantau sumber daya Amazon EC2.
- Pastikan bahwa Anda siap menangani failover. Untuk solusi dasar, Anda dapat melampirkan antarmuka jaringan atau alamat IP Elastis ke instans pengganti secara manual. Untuk informasi selengkapnya, lihat Antarmuka jaringan elastis. Untuk solusi otomatis, Anda dapat menggunakan Amazon EC2 Auto Scaling. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon EC2 Auto Scaling.
- Uji proses pemulihan instans dan volume Amazon EBS Anda secara berkala untuk memastikan bahwa data dan layanan dipulihkan dengan sukses.

Jaringan

- Tetapkan nilai time-to-live (TTL) untuk aplikasi Anda ke 255, untuk IPv4 dan. IPv6 Jika Anda menggunakan nilai yang lebih kecil, ada risiko TTL akan kedaluwarsa saat lalu lintas aplikasi sedang bergerak, sehingga menyebabkan masalah jangkauan untuk instans Anda.

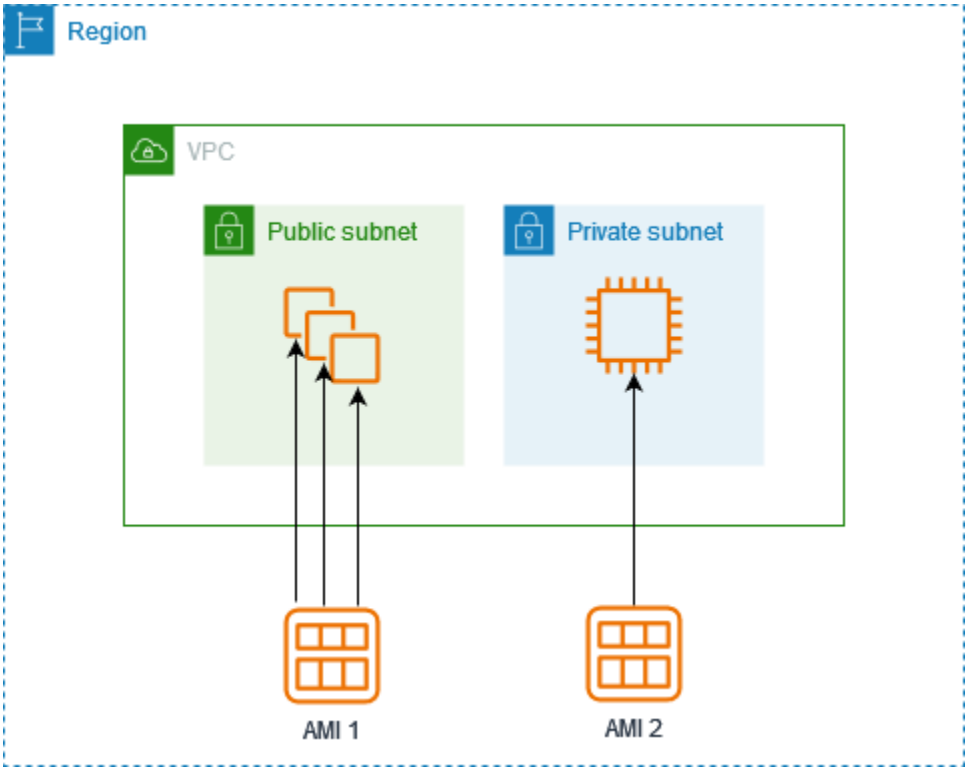
# Gambar Mesin Amazon di Amazon EC2

Amazon Machine Image (AMI) adalah gambar yang menyediakan perangkat lunak yang diperlukan untuk mengatur dan mem-boot instance Amazon EC2. Setiap AMI juga berisi pemetaan perangkat blok yang menentukan perangkat blok untuk dilampirkan ke instans yang Anda luncurkan. Anda harus menentukan AMI saat meluncurkan instans. AMI harus kompatibel dengan jenis instans yang Anda pilih untuk instance Anda. Anda dapat menggunakan AMI yang disediakan oleh AWS, AMI publik, AMI yang dibagikan orang lain kepada Anda, atau AMI yang Anda beli dari AWS Marketplace.

AMI khusus untuk hal-hal berikut:

- Region
- Sistem operasi
- Arsitektur prosesor
- Tipe volume root
- Tipe virtualisasi

Anda dapat meluncurkan beberapa instans dari satu AMI jika Anda memerlukan beberapa instans dengan konfigurasi yang sama. Anda dapat menggunakan AMI yang berbeda untuk meluncurkan instans saat Anda memerlukan instans dengan konfigurasi yang berbeda, seperti yang ditunjukkan pada diagram berikut.




Anda dapat membuat AMI dari instans Amazon EC2 dan kemudian menggunakannya untuk meluncurkan instans dengan konfigurasi yang sama. Anda dapat menyalin AMI ke Wil AWS ayah lain, dan kemudian menggunakannya untuk meluncurkan instans di Wilayah tersebut. Anda juga dapat membagikan AMI yang Anda buat dengan akun lain sehingga mereka dapat meluncurkan instans dengan konfigurasi yang sama. Anda dapat menjual AMI Anda menggunakan AWS Marketplace.

Daftar Isi


- Jenis dan karakteristik AMI di Amazon EC2
- Temukan AMI yang memenuhi persyaratan untuk instans EC2 Anda
- AMI berbayar di AWS Marketplace untuk instans Amazon EC2
- Siklus hidup AMI Amazon EC2
- Perilaku peluncuran instans dengan mode boot Amazon EC2
- Gunakan enkripsi dengan EBS-backed AMI
- Memahami penggunaan AMI bersama di Amazon EC2
- Memantau acara AMI menggunakan Amazon EventBridge
- Memahami informasi penagihan AMI
- Kuota AMI di Amazon EC2

# Jenis dan karakteristik AMI di Amazon EC2

Saat Anda meluncurkan instance, AMI yang Anda pilih harus kompatibel dengan jenis instans yang Anda pilih. Anda dapat memilih AMI untuk digunakan berdasarkan karakteristik berikut:

- Wilayah
- Sistem operasi
- Arsitektur prosesor
- Izin peluncuran
- Tipe volume root
- Tipe virtualisasi

## Izin peluncuran

Izin peluncuran menentukan siapa yang dapat menggunakan AMI untuk meluncurkan instans. Anda dapat menganggap izin peluncuran sebagai berbagi AMI — ketika Anda memberikan izin peluncuran, Anda berbagi AMI dengan pengguna lain. Hanya pemilik AMI yang dapat menentukan ketersediaannya dengan menentukan izin peluncuran. Izin peluncuran termasuk ke dalam kategori berikut.

| Izin peluncuran | Deskripsi |
|---|---|
| publik | Pemilik memberikan izin peluncuran ke semua AWS akun. |
| eksplisit | Pemilik memberikan izin peluncuran ke AWS akun, organisasi, atau unit organisasi (OU) tertentu. |
| implisit | Pemilik memiliki izin peluncuran implisit untuk AMI. |

Amazon dan komunitas Amazon EC2 menyediakan banyak pilihan AMI publik. Untuk informasi selengkapnya, lihat Memahami penggunaan AMI bersama di Amazon EC2. Developer dapat membebankan biaya untuk AMI yang mereka buat. Untuk informasi selengkapnya, lihat AMI berbayar di AWS Marketplace untuk instans Amazon EC2.

# Tipe volume root

Semua AMI dikategorikan sebagai  didukung oleh Amazon EBS  atau  didukung oleh Amazon S3.

- Amazon EBS-backed AMI — Volume root untuk instance yang diluncurkan dari AMI adalah volume Amazon Elastic Block Store (Amazon EBS) yang dibuat dari snapshot Amazon EBS. Didukung untuk AMI Linux dan Windows.
- Amazon S3-backed AMI — Volume root untuk instance yang diluncurkan dari AMI adalah volume penyimpanan instans yang dibuat dari template yang disimpan di Amazon S3. Didukung hanya untuk AMI Linux. AMI Windows tidak mendukung penyimpanan instance untuk volume root.

Untuk informasi selengkapnya, lihat Volume root untuk instans Amazon EC2 Anda.

> **Note**
>
> S3-backed AMI Amazon dianggap akhir masa pakai dan tidak direkomendasikan untuk penggunaan baru. Mereka hanya didukung pada jenis instance lama berikut: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

Tabel berikut merangkum beberapa perbedaan penting ketika menggunakan dua tipe AMI tersebut.

| Karakteristik | Amazon EBS-backed AMI | Amazon S3-backed AMI |
|---|---|---|
| Volume root | Volume EBS | Volume penyimpanan instans |
| Waktu boot untuk instans | Biasanya kurang dari 1 menit | Biasanya kurang dari 5 menit |
| Persistensi data | Secara default, volume root dihapus ketika instans berakhir.* Data di volume EBS lainnya tetap ada setelah pengakhiran instans secara default. | Data pada setiap volume penyimpanan instans hanya bertahan selama masa hidup instans. |
| Kondisi terhenti | Dapat berada dalam kondisi terhenti. Bahkan ketika instance | |

| Karakteristik | Amazon EBS-backed AMI | Amazon S3-backed AMI |
| --- | --- | --- |
| | dihentikan dan tidak berjalan, volume root tetap ada di Amazon EBS. | Tidak dapat berada dalam keadaan berhenti; instans sedang berjalan atau dihentikan. |
| Pengubahan | Tipe instans, kernel, disk RAM, dan data pengguna dapat diubah saat instans dihentikan. | Atribut instans tetap selama instan berlangsung. |
| Biaya | Anda dikenai biaya untuk penggunaan instans, penggunaa n volume EBS, dan penyimpanan AMI sebagai snapshot EBS. | Anda dikenai biaya penggunaan instans dan penyimpanan AMI di Amazon S3. |
| AMI creation/bundling | Menggunakan satu command/call | Memerlukan instalasi dan penggunaan alat AMI |

* Secara default, volume root EBS memiliki tanda `DeleteOnTermination` diatur ke `true`. Untuk informasi tentang cara mengubah tanda ini sehingga volume tetap ada setelah pengakhiran, lihat Menyimpan volume root Amazon EBS setelah instans Amazon EC2 berakhir.

** Hanya didukung dengan `io2` EBS Block Express. Untuk informasi selengkapnya, lihat Volume Blok Express SSD IOPS yang disediakan di Panduan Pengguna Amazon EBS.

## Mengidentifikasi jenis volume root yang ditentukan oleh AMI Anda

AMI yang Anda gunakan untuk meluncurkan instance EC2 menentukan jenis volume root. Volume root instans EC2 adalah volume EBS atau volume penyimpanan instans.

Nitro-based instance hanya mendukung volume root EBS. Jenis instans generasi sebelumnya berikut adalah satu-satunya jenis instans yang mendukung volume root penyimpanan instans: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

Console

Untuk mengidentifikasi tipe volume root yang ditentukan oleh AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI, lalu pilih AMI.
3. Pada tab Detail, periksa nilai Tipe perangkat root sebagai berikut:
   - `ebs`— Instans yang diluncurkan dari AMI ini akan mendapatkan volume root EBS
   - `instance store`— Instans yang diluncurkan dari AMI ini akan mendapatkan volume root penyimpanan instance.

AWS CLI

Untuk mengidentifikasi tipe volume root yang ditentukan oleh AMI

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
    --query Images[].RootDeviceType
```

Berikut ini adalah output contoh.

```
ebs
```

PowerShell

Untuk mengidentifikasi tipe volume root yang ditentukan oleh AMI

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image `
    -ImageId ami-0abcdef1234567890).RootDeviceType.Value
```

Berikut ini adalah output contoh.

```
ebs
```

# Tipe virtualisasi

Amazon Machine Image menggunakan salah satu dari dua tipe virtualisasi: paravirtual (PV) atau mesin virtual perangkat keras (HVM). Perbedaan utama antara AMI PV dan HVM adalah caranya melakukan boot dan apakah keduanya dapat memanfaatkan ekstensi perangkat keras khusus (CPU, jaringan, dan penyimpanan) untuk performa yang lebih baik. AMI Windows adalah AMI HVM.

Tabel berikut ini membandingkan AMI HVM dan PV.

| Karakteristik | HVM | PV |
| --- | --- | --- |
| Deskripsi | AMI HVM disajikan dengan set perangkat keras virtual dan boot dengan menjalankan rekaman boot master perangkat blok root gambar Anda. Tipe virtualisasi ini menyediakan kemampuan untuk menjalankan sistem operasi secara langsung pada mesin virtual tanpa modifikasi apa pun, seolah-olah dijalankan di perangkat keras bare metal. Sistem host Amazon EC2 mengemulasi beberapa atau semua perangkat keras dasar yang disajikan kepada tamu. | PV AMI melakukan boot dengan boot loader khusus yang disebut PV-GRUB, yang memulai siklus boot dan kemudian memuat rantai kernel yang ditentukan dalam `menu.lst` file pada gambar Anda. Tamu Paravirtual dapat menjalankan perangkat keras host yang tidak memiliki dukungan virtualisasi secara eksplisit. Untuk informasi selengkapnya tentang PV-GRUB dan penggunaannya di Amazon EC2, lihat Kernel yang disediakan pengguna. |
| Tipe instans yang didukung | Semua tipe instans generasi saat ini mendukung AMI HVM. | Tipe instans generasi terdahulu berikut ini mendukung AMI PV: C1, C3, M1, M3, M2, dan T1. Tipe instans generasi saat ini tidak mendukung AMI PV. |
| Dukungan untuk ekstensi perangkat keras | Tamu HVM dapat memanfaatkan ekstensi perangkat keras | Tidak, mereka tidak dapat memanfaatkan ekstensi |

| Karakteristik | HVM | PV |
| --- | --- | --- |
| | yang menyediakan akses cepat ke perangkat keras dasar di sistem host. Mereka diharuskan menggunakan jaringan yang ditingkatkan dan pemrosesan GPU. Untuk melewati instruksi ke jaringan khusus dan perangkat GPU, OS harus memiliki akses ke platform perangkat keras asli, dan virtualisasi HVM menyediakan akses ini. Untuk informasi selengkapnya, lihat Jaringan yang disempurnakan di instans Amazon EC2. | perangkat keras khusus seperti jaringan yang ditingkat kan atau pemrosesan GPU. |
| Bagaimana menemukan | Verifikasikan bahwa jenis virtualisasi AMI diatur ke `hvm`, menggunakan konsol atau perintah describe-images. | Verifikasikan bahwa jenis virtualisasi AMI diatur ke `paravirtual` , menggunak an konsol atau perintah describe-images. |

PV di HVM

Tamu paravirtual secara tradisional berkinerja lebih baik dengan operasi penyimpanan dan jaringan daripada tamu HVM karena mereka dapat memanfaatkan driver khusus untuk menghindari overhead meniru jaringan dan perangkat keras disk, sedangkan tamu HVM harus menerjemahkan instruksi ini ke perangkat keras I/O yang ditiru. Sekarang driver PV tersedia untuk tamu HVM, sehingga sistem operasi yang tidak dapat di-porting untuk dijalankan di lingkungan paravirtualisasi masih dapat melihat keunggulan kinerja dalam penyimpanan dan jaringan I/O dengan menggunakannya. Dengan PV pada driver HVM ini, tamu HVM dapat memperoleh performa yang sama, atau lebih baik daripada tamu paravirtual.

# Temukan AMI yang memenuhi persyaratan untuk instans EC2 Anda

AMI mencakup komponen dan aplikasi, seperti sistem operasi dan jenis volume root, yang diperlukan untuk meluncurkan sebuah instance. Untuk meluncurkan instance, Anda harus menemukan AMI yang memenuhi kebutuhan Anda.

Saat memilih AMI, pertimbangkan persyaratan berikut yang mungkin Anda miliki untuk instans yang ingin Anda luncurkan:

- Wil AWS ayah AMI sebagai ID AMI unik untuk setiap Wilayah.
- Sistem operasi (misalnya, Linux atau Windows).
- Arsitektur (misalnya, 32-bit, 64-bit, atau 64-bit ARM).
- Jenis volume root (misalnya, Amazon EBS atau penyimpanan instans).
- Penyedia (misalnya, Amazon Web Services).
- Perangkat lunak tambahan (misalnya, SQL Server).

Console

Anda dapat memilih dari daftar AMI saat menggunakan wizard instance peluncuran, atau Anda dapat mencari semua AMI yang tersedia menggunakan  halaman  Gambar.

Untuk menemukan AMI Mulai Cepat menggunakan wizard instance peluncuran

1. Buka konsol Amazon EC2 di  https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi, pilih Wilayah untuk meluncurkan instans Anda. Anda dapat memilih Wilayah yang tersedia untuk Anda, terlepas dari lokasi Anda. ID AMI unik untuk setiap Wil AWS ayah.
3. Dari dasbor konsol, pilih Luncurkan instans.
4. Di bawah  Aplikasi dan Gambar OS (Amazon Machine Image), pilih Mulai C  epat, pilih sistem operasi (OS) untuk instance Anda, lalu, dari  Amazon Machine Image (AMI), pilih salah satu AMI yang umum digunakan dalam daftar. Jika Anda tidak melihat AMI yang ingin Anda gunakan, pilih Telusuri AMI lainnya untuk menelusuri katalog lengkap AMI. Untuk informasi selengkapnya, lihat Aplikasi dan Gambar OS (Gambar Mesin Amazon).

Untuk menemukan AMI menggunakan halaman AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi, pilih Wilayah untuk meluncurkan instans Anda. Anda dapat memilih Wilayah yang tersedia untuk Anda, terlepas dari lokasi Anda. ID AMI unik untuk setiap Wil AWS ayah.
3. Di panel navigasi, pilih AMI.
4. (Opsional) Gunakan opsi filter dan pencarian untuk menjangkau daftar AMI yang ditampilkan untuk hanya melihat AMI yang cocok dengan kriteria Anda.

   Misalnya, untuk mencantumkan semua AMI yang disediakan oleh AWS, pilih Gambar publik. Kemudian gunakan opsi pencarian untuk memperluas daftar AMI yang ditampilkan. Pilih bilah Pencarian dan, dari menu, pilih Alias pemilik, lalu operator =, lalu nilai amazon. Untuk menemukan AMI yang cocok dengan platform tertentu, misalnya Linux atau Windows, pilih bilah Pencarian lagi untuk memilih Platform, lalu operator =, dan kemudian sistem operasi dari daftar yang disediakan.
5. (Opsional) Pilih ikon Preferensi untuk memilih atribut gambar yang akan ditampilkan, seperti jenis volume root. Atau, Anda dapat memilih AMI dari daftar dan melihat propertinya di tab Detail.
6. Sebelum memilih AMI, penting bagi Anda untuk memeriksa apakah ia didukung oleh penyimpanan instans atau Amazon EBS dan Anda mengetahui efek dari perbedaan ini. Untuk informasi selengkapnya, lihat Tipe volume root.
7. Untuk meluncurkan instans dari AMI ini, pilih dan pilih Peluncuran instans dari gambar. Untuk informasi selengkapnya tentang meluncurkan instans menggunakan konsol, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol. Jika Anda belum siap untuk meluncurkan instans, catatlah ID AMI untuk nanti.

AWS CLI

Gunakan perintah describe-images untuk menemukan AMI yang memenuhi persyaratan Anda. Secara default, perintah ini mengembalikan semua AMI yang bersifat publik, yang Anda miliki, dan yang dibagikan dengan Anda.

Untuk menemukan AMI yang dimiliki oleh Amazon

Gunakan perintah describe-images dengan opsi. `--owners`

```
aws ec2 describe-images --owners amazon
```

Untuk menemukan AMI Windows

Tambahkan filter berikut untuk hanya menampilkan AMI Windows.

```
--filters "Name=platform,Values=windows"
```

Untuk menemukan EBS-backed AMI

Tambahkan filter berikut untuk menampilkan hanya AMI yang didukung oleh Amazon EBS.

```
--filters "Name=root-device-type,Values=ebs"
```

PowerShell

Gunakan Get-EC2Image cmdlet untuk menemukan AMI yang memenuhi persyaratan Anda. Secara default, cmdlet ini mengembalikan semua AMI yang bersifat publik, yang Anda miliki, atau yang dibagikan dengan Anda.

Untuk menemukan AMI yang dimiliki oleh Amazon

Gunakan Get-EC2Image perintah dengan `-Owner` parameter.

```
Get-EC2Image -Owner amazon
```

Untuk menemukan AMI Windows

Tambahkan filter berikut untuk hanya menampilkan AMI Windows.

```
-Filter @{Name="platform"; Values="windows"}
```

Untuk contoh tambahan, lihat Menemukan Gambar Mesin Amazon Menggunakan Windows PowerShell di Panduan Alat AWS untuk PowerShell Pengguna.

Sumber daya terkait

Untuk informasi selengkapnya tentang AMI untuk sistem operasi tertentu, lihat berikut ini:

- Amazon Linux 2023 — AL2023 di Amazon EC2 di Panduan Pengguna Amazon Linux 2023
- Ubuntu — Amazon EC2 AMI Locator di situs web Canonical Ubuntu
- RHEL — Gambar Red Hat Enterprise Linux (AMI) Tersedia di Amazon Web Services (AWS) di situs web Red Hat
- Windows Server — Refer AWS ensi AMI Windows

Untuk informasi tentang AMI yang dapat Anda berlangganan di AWS Marketplace lihatAMI berbayar di AWS Marketplace untuk instans Amazon EC2.

Untuk informasi tentang menggunakan Manajer Sistem untuk membantu pengguna menemukan AMI terbaru yang harus mereka gunakan saat meluncurkan instance, lihat berikut ini:

- Referensi AMI menggunakan parameter Manajer Sistem
- Referensi AMI terbaru menggunakan parameter publik Manajer Sistem

## Referensi AMI menggunakan parameter Manajer Sistem

Saat meluncurkan instance menggunakan wizard instance peluncuran EC2 di konsol Amazon EC2, Anda dapat memilih AMI dari daftar, atau Anda dapat memilih AWS Systems Manager parameter yang menunjuk ke ID AMI (dijelaskan di bagian ini). Jika menggunakan kode automasi untuk meluncurkan instans, Anda dapat menentukan parameter Systems Manager, bukan AMI ID.

Parameter System Manager adalah pasangan nilai-kunci yang ditentukan pelanggan yang dapat Anda buat di Penyimpanan Parameter System Manager. Penyimpanan Parameter menyediakan penyimpanan pusat untuk mengeksternalisasi nilai konfigurasi aplikasi Anda. Untuk informasi selengkapnya, lihat Penyimpanan Parameter AWS dalam Panduan Pengguna AWS Systems Manager .

Ketika Anda membuat parameter yang menunjuk ke sebuah ID AMI, pastikan Anda menentukan tipe data sebagai `aws:ec2:image`. Menentukan tipe data akan memastikan bahwa ketika parameter dibuat atau dimodifikasi, nilai parameter divalidasi sebagai ID AMI. Untuk informasi selengkapnya, lihat Dukungan parameter asli untuk ID Amazon Machine Image di Panduan Pengguna AWS Systems Manager .

Daftar Isi


- Kasus penggunaan
- Izin

- Batasan
- Meluncurkan instans menggunakan parameter Systems Manager

## Kasus penggunaan

Saat Anda menggunakan parameter Systems Manager untuk menunjuk ke ID AMI, pengguna akan lebih mudah memilih AMI yang tepat saat meluncurkan instans. Parameter Systems Manager juga dapat menyederhanakan pemeliharaan kode otomatisasi.

Lebih mudah bagi pengguna

Jika suatu instans perlu diluncurkan menggunakan AMI tertentu, dan AMI diperbarui secara rutin, kami sarankan Anda meminta pengguna untuk memilih parameter Systems Manager untuk mencari AMI. Mengharuskan pengguna memilih parameter Systems Manager memastikan AMI terbaru digunakan untuk meluncurkan instans.

Sebagai contoh, setiap bulan di organisasi, Anda dapat membuat versi AMI baru yang memiliki sistem operasi dan patch aplikasi terbaru. Anda juga memerlukan pengguna untuk meluncurkan instans menggunakan AMI versi terbaru Anda. Untuk memastikan pengguna menggunakan versi terbaru, Anda dapat membuat parameter Systems Manager (misalnya, `golden-ami`) yang menunjuk ke ID AMI yang benar. Setiap kali versi baru AMI dibuat, Anda memperbarui nilai ID AMI di parameter sehingga selalu mengarah ke AMI terbaru. Pengguna tidak perlu mengetahui pembaruan berkala untuk AMI karena mereka terus memilih parameter Systems Manager yang sama setiap saat. Penggunaan parameter Systems Manager untuk AMI Anda memudahkan mereka dalam memilih AMI yang benar untuk peluncuran instans.

Menyederhanakan pemeliharaan kode automasi

Jika menggunakan kode automasi untuk meluncurkan instans, Anda dapat menentukan parameter Systems Manager, bukan ID AMI. Setiap kali versi baru AMI dibuat, Anda memperbarui nilai ID AMI di parameter sehingga selalu mengarah ke AMI terbaru. Kode otomatisasi yang mengacu pada parameter tidak harus dimodifikasi setiap kali versi baru AMI dibuat. Hal ini menyederhanakan pemeliharaan otomatisasi dan membantu menurunkan biaya deployment.

> **Note**
>
> Instans yang berjalan tidak terpengaruh saat Anda mengubah ID AMI yang ditunjuk oleh parameter Systems Manager.

### Izin

Jika Anda menggunakan parameter Manajer Sistem yang mengarah ke ID AMI di wizard instance peluncuran, Anda harus menambahkan izin berikut ke kebijakan IAM Anda:

- `ssm:DescribeParameters`— Memberikan izin untuk melihat dan memilih parameter Manajer Sistem.
- `ssm:GetParameters`- Memberikan izin untuk mengambil nilai-nilai parameter Manajer Sistem.

Anda juga dapat membatasi akses ke parameter Systems Manager tertentu. Untuk informasi selengkapnya dan contoh kebijakan IAM, lihatContoh: Menggunakan wizard peluncuran instans EC2.

### Batasan

AMI dan parameter Systems Manager bersifat khusus Wilayah. Untuk menggunakan nama parameter Systems Manager yang sama di seluruh Wilayah, buatlah parameter Systems Manager di setiap Wilayah dengan nama yang sama (misalnya, `golden-ami`). Di setiap Wilayah, arahkan parameter Systems Manager ke AMI di dalam Wilayah tersebut.

Nama parameter peka huruf besar/kecil. Garis miring terbalik untuk nama parameter hanya diperlukan jika parameter adalah bagian dari hierarki, misalnya, `/amis/production/golden-ami`. Anda dapat menghilangkan garis miring terbalik jika parameter bukan bagian dari hirarki.

### Meluncurkan instans menggunakan parameter Systems Manager

Saat meluncurkan instance, alih-alih menentukan ID AMI, Anda dapat menentukan parameter Manajer Sistem yang menunjuk ke ID AMI.

Untuk menentukan parameter secara terprogram, gunakan sintaks berikut, di mana `resolve:ssm` adalah awalan standar dan `parameter-name` merupakan nama parameter unik.

```
resolve:ssm:parameter-name
```

Parameter Systems Manager memiliki dukungan versi. Setiap iterasi parameter diberi nomor versi unik. Anda dapat mereferensikan versi parameter sebagai berikut, di mana `version` adalah nomor versi unik. Secara default, versi terbaru parameter digunakan ketika tidak ada versi yang ditentukan.

```
resolve:ssm:parameter-name:version
```

Untuk meluncurkan instance menggunakan parameter publik yang disediakan oleh AWS, lihatReferensi AMI terbaru menggunakan parameter publik Manajer Sistem.

Console

Untuk menemukan AMI menggunakan parameter Manajer Sistem

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi, pilih Wilayah untuk meluncurkan instans Anda. Anda dapat memilih Wilayah yang tersedia untuk Anda, terlepas dari lokasi Anda.
3. Dari dasbor konsol, pilih Luncurkan instans.
4. Di bawah Gambar Aplikasi dan OS (Amazon Machine Image), pilih Telusuri AMI lainnya.
5. Pilih tombol panah di sebelah kanan bilah pencarian, lalu pilih parameter Pencarian dengan Systems Manager.
6. Untuk Parameter System Manager, pilih parameter. ID AMI terkait akan muncul di bawah Saat ini memutuskan ke.
7. Pilih Pencarian. AMI yang cocok dengan ID AMI muncul dalam daftar.
8. Pilih AMI dari daftar, lalu pilih Pilih.

Untuk informasi tentang peluncuran instans menggunakan wizard peluncuran instans, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

AWS CLI

Untuk meluncurkan instance menggunakan parameter Manajer Sistem

Gunakan perintah run-instance dengan opsi. `--image-id` Contoh ini menggunakan parameter Manajer Sistem bernama**`golden-ami`**, yang menentukan ID AMI.

```
--image-id resolve:ssm:/golden-ami
```

Anda dapat membuat versi parameter Manajer Sistem. Contoh berikut menentukan versi 2 dari **`golden-ami`** parameter.

```
--image-id resolve:ssm:/golden-ami:2
```

PowerShell

Untuk meluncurkan instance menggunakan parameter Manajer Sistem

Gunakan New-EC2Instance cmdlet dengan parameter. `-ImageId` Contoh ini menggunakan parameter Manajer Sistem bernama**`golden-ami`**, yang menentukan ID AMI.

```
-ImageId "resolve:ssm:/golden-ami"
```

Anda dapat membuat versi parameter Manajer Sistem. Contoh berikut menentukan versi 2 dari **`golden-ami`** parameter.

```
-ImageId "resolve:ssm:/golden-ami:2"
```

## Referensi AMI terbaru menggunakan parameter publik Manajer Sistem

AWS Systems Manager menyediakan parameter publik untuk AMI publik yang dikelola oleh AWS. Anda dapat menggunakan parameter publik saat meluncurkan instans untuk memastikan bahwa Anda menggunakan AMI terbaru. Misalnya, parameter publik `/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-arm64` tersedia di semua Wilayah dan selalu menunjuk ke versi terbaru dari Amazon Linux 2023 AMI untuk arsitektur arm64 di Wilayah tertentu.

Parameter publik tersedia dari jalur berikut:

- Linux – `/aws/service/ami-amazon-linux-latest`
- Windows – `/aws/service/ami-windows-latest`

Untuk informasi selengkapnya, lihat Bekerja dengan parameter publik di Panduan Pengguna AWS Systems Manager .

### Temukan parameter SSM untuk AMI publik

Saat Anda mendeskripsikan AMI publik, respons menyertakan parameter terkait AWS Systems Manager (SSM) di `PublicSsmParameterName` bidang. Gunakan parameter ini untuk mengidentifikasi parameter SSM yang selalu menunjuk ke versi terbaru AMI tersebut.

Console

Untuk menemukan parameter SSM untuk AMI publik

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.

3. Pilih AMI publik.
4. Pada tab Detail, temukan nama parameter SSM Publik.

AWS CLI

Untuk menemukan parameter SSM untuk AMI publik

Gunakan perintah describe-images. Respons mencakup `PublicSsmParameterName` bidang untuk AMI publik yang memiliki parameter SSM terkait.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890
```

Contoh output berikut menunjukkan `PublicSsmParameterName` bidang. Beberapa bidang dihilangkan untuk singkatnya.

```
{
    "Images": [
        {
            "ImageId": "ami-0abcdef1234567890",
            "Name": "al2023-ami-2023.7.20260601.0-kernel-6.1-x86_64",
            ...
            "PublicSsmParameterName": "aws/service/ami-amazon-linux-latest/al2023-
ami-kernel-default-x86_64"
        }
    ]
}
```

PowerShell

Untuk menemukan parameter SSM untuk AMI publik

Gunakan Get-EC2Image cmdlet. Respons mencakup `PublicSsmParameterName` bidang untuk AMI publik yang memiliki parameter SSM terkait.

```
Get-EC2Image -ImageId ami-0abcdef1234567890
```

Contoh output berikut menunjukkan `PublicSsmParameterName` bidang. Beberapa bidang dihilangkan untuk singkatnya.

```
ImageId                : ami-0abcdef1234567890
Name                   : al2023-ami-2023.7.20260601.0-kernel-6.1-x86_64
...
PublicSsmParameterName : aws/service/ami-amazon-linux-latest/al2023-ami-kernel-
default-x86_64
```

## Daftar AMI Amazon Linux

### AWS CLI

Untuk membuat daftar AMI Linux saat ini AWS Region

Gunakan perintah get-parameter s-by-path berikut. Nilai untuk `--path` parameter khusus untuk AMI Linux.

```
aws ssm get-parameters-by-path \
    --path /aws/service/ami-amazon-linux-latest \
    --query "Parameters[].Name"
```

### PowerShell

Untuk membuat daftar AMI Linux saat ini AWS Region

Gunakan Get-SSMParametersByPath cmdlet.

```
Get-SSMParametersByPath `
    -Path "/aws/service/ami-amazon-linux-latest" | `
    Sort-Object Name | Format-Table Name
```

## Daftar AMI Windows

### AWS CLI

Untuk mencantumkan AMI Windows di saat ini AWS Region

Gunakan perintah get-parameter s-by-path berikut. Nilai untuk `--path` parameter khusus untuk AMI Windows.

```
aws ssm get-parameters-by-path \
```

```
    --path /aws/service/ami-windows-latest \
    --query "Parameters[].Name"
```

PowerShell

Untuk mencantumkan AMI Windows di saat ini AWS Region

Gunakan Get-SSMParametersByPath cmdlet.

```
Get-SSMParametersByPath `
    -Path "/aws/service/ami-windows-latest" | `
    Sort-Object Name | Format-Table Name
```

## Luncurkan instance menggunakan parameter publik

Untuk menentukan parameter publik saat meluncurkan instance, gunakan sintaks berikut:`resolve:ssm:`*`public-parameter`*, di mana `resolve:ssm` awalan standar dan *`public-parameter`* merupakan jalur dan nama parameter publik.

AWS CLI

Untuk meluncurkan suatu instans menggunakan parameter publik

Gunakan perintah run-instance dengan opsi. `--image-id` Contoh ini menentukan parameter publik Manajer Sistem untuk ID gambar untuk meluncurkan instance menggunakan AMI Amazon Linux 2023 terbaru

```
--image-id resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64
```

PowerShell

Untuk meluncurkan suatu instans menggunakan parameter publik

Gunakan New-EC2Instance cmdlet dengan parameter. `-ImageId` Contoh ini menentukan parameter publik Manajer Sistem untuk ID gambar untuk meluncurkan instance menggunakan AMI terbaru untuk Windows Server 2022.

```
-ImageId "resolve:ssm:/aws/service/ami-windows-latest/Windows_Server-2022-English-Full-Base"
```

Untuk contoh selengkapnya yang menggunakan parameter Manajer Sistem, lihat Kueri ID AMI Amazon Linux terbaru Menggunakan Penyimpanan AWS Systems Manager Parameter dan Ku eri untuk AMI Windows Terbaru Menggunakan Peny AWS Systems Manager impanan Parameter.

# AMI berbayar di AWS Marketplace untuk instans Amazon EC2

AMI ber bayar adalah AMI yang terdaftar untuk dijual di AWS Marketplace. Ini AWS Marketplace adalah toko online tempat Anda dapat membeli perangkat lunak yang berjalan AWS, termasuk AMI yang dapat Anda gunakan untuk meluncurkan instans EC2 Anda. AWS Marketplace AMI diatur ke dalam kategori, seperti Alat Pengembang, untuk memungkinkan Anda menemukan produk yang sesuai dengan kebutuhan Anda. Untuk informasi lebih lanjut tentang AWS Marketplace, lihat situs AWS Marketplace web.

Anda dapat membeli AMI AWS Marketplace dari pihak ketiga, termasuk AMI yang datang dengan kontrak layanan dari organisasi seperti Red Hat. Anda juga dapat membuat AMI dan menjualnya AWS Marketplace ke pengguna Amazon EC2 lainnya. Membangun AMI yang aman, terjaga, dan dapat digunakan untuk konsumsi publik adalah hal yang cukup mudah jika Anda mengikuti beberapa langkah panduan sederhana. Untuk informasi tentang cara membuat dan menggunakan AMI bersama, lihat Memahami penggunaan AMI bersama di Amazon EC2.

Meluncurkan instans dari AMI berbayar sama seperti meluncurkan instans dari AMI lainnya. Tidak perlu parameter tambahan. Instans dikenakan biaya sesuai dengan tarif yang ditetapkan oleh pemilik AMI, serta biaya penggunaan standar untuk layanan web terkait, misalnya, tarif per jam untuk menjalankan jenis instans m5.small di Amazon EC2. Pajak tambahan mungkin juga berlaku. Pemilik AMI berbayar dapat mengonfirmasi apakah instans tertentu diluncurkan menggunakan AMI berbayar tersebut.

> **Important**
>
> Amazon DevPay tidak lagi menerima penjual atau produk baru. AWS Marketplace Sekarang adalah platform e-commerce tunggal terpadu untuk menjual perangkat lunak dan layanan melalui AWS. Untuk informasi tentang cara menerapkan dan menjual perangkat lunak dari AWS Marketplace, lihat Men jual di AWS Marketplace. AWS Marketplace mendukung AMI yang didukung oleh Amazon EBS.

Daftar Isi

- Jual AMI Anda di AWS Marketplace
- Mencari AMI berbayar
- Beli AMI berbayar di AWS Marketplace
- Ambil AWS Marketplace kode produk dari instance Anda
- Gunakan dukungan berbayar untuk dukungan AWS Marketplace persembahan
- Tagihan untuk AMI berbayar dan didukung
- Kelola Anda AWS Marketplace langganan


## Jual AMI Anda di AWS Marketplace

Anda dapat menjual AMI Anda menggunakan AWS Marketplace. AWS Marketplace menawarkan pengalaman belanja yang terorganisir. Selain itu, AWS Marketplace juga mendukung AWS fitur seperti Amazon EBS-backed AMI, Instans Cadangan, dan Instans Spot.

Untuk informasi tentang cara menjual AMI Anda di AWS Marketplace, lihat Men jual di AWS Marketplace.

## Mencari AMI berbayar

AMI berbayar adalah Amazon Machine Image (AMI) yang tersedia untuk dibeli. AMI berbayar juga memiliki kode produk. Anda dapat menemukan AMI yang tersedia untuk dibeli di AWS Marketplace.

Console

Untuk menemukan AMI berbayar

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih Gambar publik untuk filter pertama.
4. Lakukan salah satu tindakan berikut:
   - Jika Anda mengetahui kode produk, pilih Kode Produk, lalu =, lalu masukkan kode produk.
   - Jika Anda tidak tahu kode produk, di bilah Pencarian, tentukan filter berikut: Alias pemilik = aws-marketplace. Tentukan filter tambahan sesuai kebutuhan.
5. Simpan ID AMI.

AWS CLI

Untuk menemukan AMI berbayar

Gunakan perintah describe-images berikut.

```
aws ec2 describe-images --owners aws-marketplace
```

Outputnya mencakup sejumlah besar gambar. Anda dapat menentukan filter untuk membantu Anda menentukan AMI mana yang Anda butuhkan. Setelah Anda menemukan AMI, tentukan ID-nya dalam perintah berikut untuk mendapatkan kode produknya.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
    --query Images[*].ProductCodes[].ProductCodeId
```

Berikut ini adalah output contoh.

```
[
    "cdef1234abc567def8EXAMPLE"
]
```

Jika Anda mengetahui kode produk, Anda dapat memfilter hasilnya berdasarkan kode produk. Contoh berikut ini memberikan AMI terbaru dengan kode produk tertentu.

```
aws ec2 describe-images \
    --filters "Name=product-code,Values=cdef1234abc567def8EXAMPLE" \
    --query "sort_by(Images, &CreationDate)[-1].[ImageId]"
```

PowerShell

Untuk menemukan AMI berbayar

Gunakan Get-EC2Image cmdlet.

```
Get-EC2Image -Owner aws-marketplace
```

Outputnya mencakup sejumlah besar gambar. Anda dapat menentukan filter untuk membantu Anda menentukan AMI mana yang Anda butuhkan. Setelah Anda menemukan AMI, tentukan ID-nya dalam perintah berikut untuk mendapatkan kode produknya.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).ProductCodes
```

Berikut ini adalah output contoh.

```
ProductCodeId             ProductCodeType
-------------             ---------------
cdef1234abc567def8EXAMPLE marketplace
```

Jika Anda mengetahui kode produk, Anda dapat memfilter hasilnya berdasarkan kode produk. Contoh berikut ini memberikan AMI terbaru dengan kode produk tertentu.

```
(Get-EC2Image -Owner aws-marketplace -Filter @{"Name"="product-
code";"Value"="cdef1234abc567def8EXAMPLE"} | sort CreationDate -Descending | Select-
Object -First 1).ImageId
```

## Beli AMI berbayar di AWS Marketplace

Anda harus mendaftar untuk (membeli) AMI berbayar sebelum dapat meluncurkan instans Amazon EC2 menggunakan AMI.

Biasanya, penjual AMI berbayar memberi Anda informasi tentang AMI, termasuk harga dan tautan tempat Anda dapat membelinya. Ketika Anda memilih tautan, pertama-tama Anda diminta untuk masuk AWS, dan kemudian Anda dapat membeli AMI.

### Membeli AMI berbayar menggunakan konsol

Anda dapat membeli AMI berbayar dengan menggunakan wizard peluncuran Amazon EC2. Untuk informasi selengkapnya, lihat Luncurkan instans Amazon EC2 dari AMI AWS Marketplace.

### Berlangganan produk menggunakan AWS Marketplace

Untuk menggunakan AWS Marketplace, Anda harus memiliki Akun AWS. Untuk meluncurkan instans dari AWS Marketplace produk, Anda harus mendaftar untuk menggunakan layanan Amazon EC2, dan Anda harus berlangganan produk untuk meluncurkan instans. Anda dapat menggunakan salah satu metode berikut untuk berlangganan produk di AWS Marketplace:

- AWS Marketplace situs web: Anda dapat meluncurkan perangkat lunak yang telah dikonfigurasi dengan cepat dengan fitur penyebaran 1-Klik. Untuk informasi lebih lanjut, lihat AMI-based produk di AWS Marketplace.

- Wizard peluncuran Amazon EC2: Anda dapat mencari AMI dan meluncurkan instans langsung dari wizard. Untuk informasi selengkapnya, lihat Luncurkan instans Amazon EC2 dari AMI AWS Marketplace.

## Ambil AWS Marketplace kode produk dari instance Anda

Anda dapat mengambil kode AWS Marketplace produk untuk instance Anda menggunakan metadata instansnya. Jika instans memiliki kode produk, Amazon EC2 akan memberikannya. Untuk informasi selengkapnya tentang pengambilan metadata instans, lihat Mengakses metadata instans untuk instans EC2.

IMDSv2

Linux

Jalankan perintah berikut dari instance Linux Anda.

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/product-codes
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda.

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
    -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
    -Method GET -Uri http://169.254.169.254/latest/meta-data/product-codes
```

IMDSv1

Linux

Jalankan perintah berikut dari instance Linux Anda.

```
curl http://169.254.169.254/latest/meta-data/product-codes
```

Windows

Jalankan perintah berikut dari instance Windows Anda.

```
Invoke-RestMethod -Uri http://169.254.169.254/latest/meta-data/product-codes
```

## Gunakan dukungan berbayar untuk dukungan AWS Marketplace persembahan

Amazon EC2 juga memungkinkan developer menawarkan dukungan untuk perangkat lunak (atau AMI turunannya). Developer dapat membuat produk dukungan yang dapat Anda gunakan dengan mendaftar. Saat mendaftar untuk produk dukungan, developer memberi Anda kode produk, yang kemudian harus dikaitkan dengan AMI Anda sendiri. Hal ini memungkinkan developer untuk mengonfirmasi bahwa instans Anda memenuhi syarat untuk dukungan. Ini juga memastikan ketika Anda menjalankan instans produk, Anda dikenai biaya sesuai ketentuan untuk produk tertentu dari developer.

Batasan

- Setelah Anda menetapkan atribut kode produk, atribut tersebut tidak dapat diubah atau dihapus.
- Anda tidak dapat menggunakan produk dukungan dengan Instans Tercadang. Anda selalu membayar harga yang ditetapkan oleh penjual produk dukungan.

AWS CLI

Untuk mengaitkan kode produk dengan AMI Anda

Gunakan perintah modify- image-atribut.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --product-codes "cdef1234abc567def8EXAMPLE"
```

PowerShell

Untuk mengaitkan kode produk dengan AMI Anda

Gunakan Edit-EC2ImageAttribute cmdlet.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -ProductCode "cdef1234abc567def8EXAMPLE"
```

## Tagihan untuk AMI berbayar dan didukung

Di setiap akhir bulan, Anda akan menerima email dengan jumlah yang ditagih ke kartu kredit untuk penggunaan AMI yang dibayar atau didukung selama bulan tersebut. Tagihan ini terpisah dari tagihan Amazon EC2 reguler Anda. Untuk informasi selengkapnya, lihat Membayar produk dalam Panduan Pembeli AWS Marketplace .

## Kelola Anda AWS Marketplace langganan

Di AWS Marketplace situs web, Anda dapat memeriksa detail langganan Anda, melihat petunjuk penggunaan vendor, mengelola langganan Anda, dan banyak lagi.

### Periksa detail langganan

Untuk memeriksa detail langganan Anda

1. Masuk ke AWS Marketplace.
2. Pilih Akun Marketplace Anda.
3. Pilih Kelola langganan perangkat lunak Anda.
4. Semua langganan Anda saat ini akan tercantum. Pilih  Petunjuk Penggunaan  untuk melihat petunjuk khusus untuk menggunakan produk, misalnya, nama pengguna untuk menghubungkan ke instans yang sedang berjalan.

### Batalkan langganan

> **Note**
>
> - Membatalkan langganan tidak menghentikan instans yang diluncurkan dengan AMI tersebut. Kami akan terus menagih Anda untuk instans yang sedang berjalan hingga instans tersebut dihentikan. Anda harus menghentikan semua instans yang diluncurkan dengan AMI untuk menghentikan penagihan untuk langganan.

- Setelah membatalkan langganan, Anda tidak lagi dapat meluncurkan instans apa pun dari AMI tersebut. Untuk menggunakan AMI itu lagi, Anda harus berlangganan kembali, baik di situs AWS Marketplace web, atau melalui wizard peluncuran di konsol Amazon EC2.

Untuk membatalkan AWS Marketplace langganan

1. Untuk menghentikan penagihan untuk langganan, pastikan Anda telah menghentikan instans apa pun yang berjalan dari langganan.

   > **Warning**
   >
   > Mengakhiri instance bersifat permanen dan tidak dapat diubah.
   > Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
   > Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

   a. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
   b. Di panel navigasi, pilih Instans.
   c. Pilih instance, lalu pilih Status instans, H entikan (hapus) instance.
   d. Pilih Hentikan (hapus) saat diminta konfirmasi.
2. Masuk ke AWS Marketplace, dan pilih Akun Marketplace Anda, lalu Kelola langganan perangkat lunak Anda.
3. Pilih Batalkan langganan. Anda diminta untuk mengonfirmasi pembatalan.

# Siklus hidup AMI Amazon EC2

Amazon Machine Image (AMI) adalah gambar yang berisi konfigurasi perangkat lunak yang diperlukan untuk mengatur dan mem-boot instance. Anda harus menentukan AMI saat meluncurkan instans. Anda dapat menggunakan AMI yang disediakan oleh Amazon, atau Anda dapat membuat AMI Anda sendiri. AMI harus berada Wilayah AWS di tempat Anda ingin meluncurkan instans Anda.

Siklus hidup AMI mencakup pembuatan, penyalinan, penghentian, penonaktifan, dan penghapusan (deregistrasi) AMI.

Buat AMI. Meskipun Amazon menyediakan AMI yang dapat Anda gunakan untuk meluncurkan instans, Anda dapat membuat AMI khusus yang disesuaikan dengan kebutuhan Anda. Untuk membuat AMI khusus, luncurkan instance dari AMI yang ada, sesuaikan instance (misalnya, instal perangkat lunak dan konfigurasikan pengaturan sistem operasi), lalu buat AMI dari instance. Setiap penyesuaian instans disimpan ke AMI baru, sehingga instans yang diluncurkan dari AMI baru Anda menyertakan penyesuaian ini.

AMI yang dapat dibuktikan. Untuk membuat AMI yang mendukung pengesahan instans EC2, lihat. AMI yang dapat dibuktikan

Salin AMI. Anda dapat menggunakan AMI untuk meluncurkan instance hanya Wilayah AWS di mana AMI berada. Jika Anda perlu meluncurkan instans dengan konfigurasi yang sama di beberapa Wilayah, salin AMI ke Wilayah lainnya.

Hentikan AMI. Untuk menandai AMI sebagai tergantikan atau kedaluwarsa, Anda dapat menetapkan tanggal penghentian segera atau di masa mendatang. AMI yang tidak digunakan lagi disembunyikan dari daftar AMI, tetapi pengguna dan layanan dapat terus menggunakan AMI yang tidak digunakan lagi jika mereka mengetahui ID AMI.

Nonaktifkan AMI. Untuk sementara mencegah AMI digunakan, Anda dapat menonaktifkannya. Ketika AMI dinonaktifkan, itu tidak dapat digunakan untuk meluncurkan instans baru. Namun, jika Anda mengaktifkan kembali AMI, ini dapat digunakan untuk meluncurkan instans lagi. Perhatikan bahwa menonaktifkan AMI tidak memengaruhi instans yang sudah ada yang telah diluncurkan darinya.

Batalkan pendaftaran (hapus) AMI. Ketika Anda tidak lagi memerlukan AMI, Anda dapat membatalkan pendaftaran, mencegahnya digunakan untuk meluncurkan instans baru. Jika AMI cocok dengan aturan retensi, ia akan pindah ke Recycle Bin, di mana AMI dapat dipulihkan sebelum periode retensinya berakhir, setelah itu dihapus secara permanen. Jika tidak cocok dengan aturan retensi, aturan tersebut akan dihapus secara permanen segera. Perhatikan bahwa membatalkan pendaftaran AMI tidak memengaruhi instans yang ada yang diluncurkan dari AMI.

Mengotomatiskan siklus hidup AMI. Anda dapat menggunakan Amazon Data Lifecycle Manager untuk mengotomatiskan pembuatan, penyimpanan, penyalinan, penghentian, dan deregistrasi Amazon EBS-backed AMI dan snapshot pendukungnya. Anda juga dapat menggunakan EC2 Image Builder untuk mengotomatiskan pembuatan, pengelolaan, penyebaran AMI yang disesuaikan. Untuk

informasi selengkapnya, lihat Mengotomatiskan pencadangan dengan Amazon Data Lifecycle Manager di Panduan Pengguna Amazon EBS dan Panduan Pengguna EC2 Image Builder.

Daftar Isi


- Buat EBS-backed AMI Amazon
- Buat S3-backed AMI Amazon
- Buat AMI Amazon EC2 menggunakan Windows Sysprep
- Salin AMI Amazon EC2
- Simpan dan pulihkan AMI menggunakan S3
- Gunakan nenek moyang AMI untuk melacak asal usul AMI
- Gunakan tanda air AMI untuk melacak dan mengidentifikasi AMI
- Jenis instans yang diizinkan AMI
- Mengelola dan pantau penggunaan AMI
- Menghentikan penggunaan AMI Amazon EC2
- Nonaktifkan AMI Amazon EC2
- Batalkan pendaftaran ke Amazon EC2 AMI


## Buat EBS-backed AMI Amazon

Anda dapat membuat Amazon EBS-backed AMI Anda sendiri dari instans Amazon EC2 atau dari snapshot volume root instans Amazon EC2.

Untuk membuat Amazon EBS-backed AMI dari instance, mulailah dengan meluncurkan instance menggunakan EBS-backed AMI Amazon yang ada. AMI ini dapat menjadi yang Anda peroleh dari AWS Marketplace, dibuat menggunakan VM Import/Export, atau AMI lain yang dapat Anda akses. Setelah menyesuaikan instance untuk memenuhi persyaratan spesifik Anda, buat dan daftarkan AMI baru. Anda kemudian dapat menggunakan AMI baru untuk meluncurkan instans baru dengan penyesuaian Anda.

> **Note**
>
> Untuk membuat AMI yang mendukung pengesahan instans EC2, lihat. AMI yang dapat dibuktikan

Prosedur yang dijelaskan di bawah berfungsi untuk instans Amazon EC2 yang didukung oleh volume Amazon Elastic Block Store (Amazon EBS) terenkripsi (termasuk volume root) serta volume yang tidak dienkripsi.

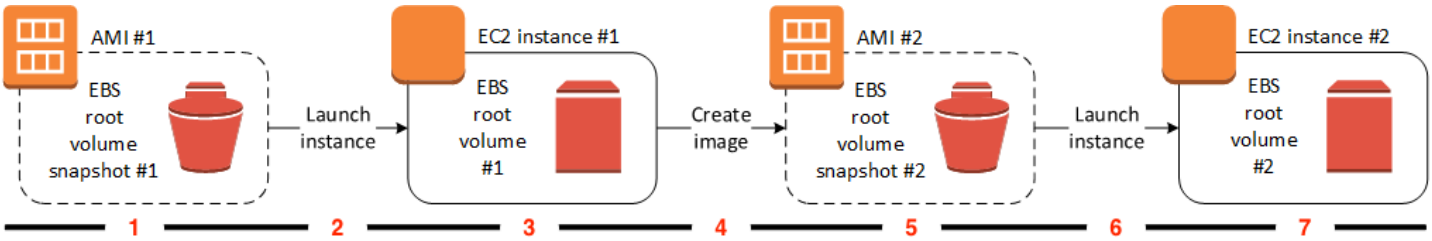
Proses pembuatan AMI berbeda untuk S3-backed AMI Amazon. Untuk informasi selengkapnya, lihat Buat S3-backed AMI Amazon.

Daftar Isi


- Ikhtisar pembuatan AMI dari instance
- Membuat AMI dari instance
- Buat AMI dari snapshot


## Ikhtisar pembuatan AMI dari instance

Diagram berikut merangkum proses untuk membuat Amazon EBS-backed AMI dari instans EC2 yang sedang berjalan: Mulailah dengan AMI yang ada, luncurkan instance, sesuaikan, buat AMI baru darinya, dan akhirnya luncurkan instance AMI baru Anda. Angka-angka dalam diagram cocok dengan angka-angka dalam deskripsi berikut.




1 – AMI #1: Mulailah dengan AMI yang ada

Cari AMI yang mirip dengan AMI yang ingin Anda buat. Ini bisa berupa AMI yang telah Anda peroleh dari AWS Marketplace, AMI yang telah Anda buat menggunakan VM Import/Export, atau AMI lain yang dapat Anda akses. Anda akan mengkustomisasi AMI ini untuk kebutuhan Anda.

Dalam diagram, snapshot volume akar EBS #1 menunjukkan bahwa AMI adalah EBS-backed AMI Amazon dan informasi tentang volume root disimpan dalam snapshot ini.

2 – Luncurkan instans dari AMI yang ada

Cara mengonfigurasi AMI adalah dengan meluncurkan instans dari AMI tempat Anda ingin mendasarkan AMI baru Anda, lalu mengustomisasi instans (ditunjukkan pada 3 dalam diagram). Kemudian, Anda akan membuat AMI baru yang menyertakan kustomisasi (ditunjukkan pada 4 dalam diagram).

3 – Instans EC2 #1: Kustomisasi instans

Sambungkan ke instans Anda dan kustomisasi sesuai kebutuhan. AMI Anda yang baru akan menyertakan kustomisasi ini.

Anda dapat melakukan tindakan berikut ini pada instans Anda untuk mengustomisasikannya:

- Menginstal perangkat lunak dan aplikasi
- Menyalin data
- Kurangi waktu mulai dengan menghapus file sementara dan mendefragmentasi hard drive Anda
- Melampirkan volume EBS tambahan

4 – Buat gambar

Saat Anda membuat AMI dari instans, Amazon EC2 menonaktifkan instans sebelum membuat AMI guna memastikan segala sesuatu yang ada pada instans dihentikan dan dalam kondisi yang konsisten selama proses pembuatan. Jika Anda yakin instans Anda berada dalam kondisi konsisten yang cocok untuk pembuatan AMI, Anda dapat memberi tahu Amazon EC2 untuk tidak mematikan dan melakukan boot ulang instans. Beberapa sistem file, seperti XFS, dapat membekukan dan membatalkan pembekuan aktivitas sehingga aman untuk membuat gambar tanpa boot ulang instans.

Selama AMI-creation proses, Amazon EC2 membuat snapshot dari volume root instans Anda dan volume EBS lainnya yang dilampirkan ke instans Anda. Anda dikenai biaya untuk snapshot hingga Anda membatalkan pendaftaran AMI dan menghapus snapshot. Jika setiap volume yang dilampirkan pada instans dienkripsi, AMI yang baru hanya akan berhasil diluncurkan pada tipe instans yang mendukung enkripsi Amazon EBS.

Tergantung pada ukuran volume, dibutuhkan beberapa menit untuk menyelesaikan AMI-creation proses (kadang-kadang hingga 24 jam). Anda mungkin merasa lebih efisien untuk membuat snapshot volume sebelum membuat AMI. Dengan begitu, hanya snapshot kecil dan bertahap yang perlu dibuat saat AMI dibuat, dan proses ini selesai lebih cepat (total waktu untuk pembuatan snapshot tetap sama).

5 – AMI #2: AMI Baru

Setelah proses selesai, Anda memiliki AMI baru dan snapshot (snapshot #2) dibuat dari volume root instans. Jika Anda menambahkan volume penyimpanan instans atau volume EBS ke instans, selain volume root, pemetaan perangkat blok untuk AMI baru berisi informasi untuk volume ini.

Amazon EC2 secara otomatis mendaftarkan AMI untuk Anda.

6 – Meluncurkan instans dari AMI baru

Anda dapat menggunakan AMI baru untuk meluncurkan instans.

7 – Instans EC2 #2: Instans baru

Saat Anda meluncurkan instans menggunakan AMI baru, Amazon EC2 membuat volume EBS baru untuk volume root instans menggunakan snapshot. Jika Anda menambahkan volume penyimpanan instans atau volume EBS saat Anda menyesuaikan instance, pemetaan perangkat blok untuk AMI baru berisi informasi untuk volume ini, dan pemetaan perangkat blok untuk instans yang Anda luncurkan dari AMI baru secara otomatis berisi informasi untuk volume ini. Volume penyimpanan instans yang ditentukan dalam pemetaan perangkat blok untuk instans yang baru adalah volume baru dan tidak berisi data dari volume penyimpanan instans yang Anda gunakan untuk membuat AMI. Data di volume EBS tetap ada. Untuk informasi selengkapnya, lihat Memblokir pemetaan perangkat untuk volume pada instans Amazon EC2.

Saat Anda membuat instance baru dari EBS-backed AMI, Anda harus menginisialisasi volume root dan penyimpanan EBS tambahan sebelum memasukkannya ke dalam produksi. Untuk informasi selengkapnya, lihat Menginisialisasi volume Amazon EBS di Panduan Pengguna Amazon EBS.

## Membuat AMI dari instance

Jika Anda memiliki instance yang sudah ada, Anda dapat membuat AMI dari instance ini.

Console

Untuk membuat AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans untuk membuat AMI, lalu pilih Tindakan, Gambar dan templat, Buat gambar.

> **Tip**
>
> Jika opsi ini dinonaktifkan, instans Anda bukan EBS-backed instans Amazon.

4. Pada halaman Buat gambar, tentukan informasi berikut:

   a. Untuk Nama gambar, masukkan nama yang unik untuk gambar, hingga 127 karakter.

b. Untuk Deskripsi gambar, masukkan deskripsi opsional gambar, hingga 255 karakter.

c. Untuk contoh Reboot, tetap pilih kotak centang (default), atau hapus.

- Jika instans Re boot dipilih, saat Amazon EC2 membuat AMI baru, instans akan di-boot ulang sehingga dapat mengambil snapshot dari volume yang dilampirkan saat data diam, untuk memastikan status yang konsisten.
- Jika instans Reboot dihapus, saat Amazon EC2 membuat AMI baru, instans tidak akan dimatikan dan me-reboot.

> ⚠ Warning
>
> Jika Anda menghapus instance Reboot, kami tidak dapat menjamin integritas sistem file dari gambar yang dibuat.

d. Volume instans – Anda dapat memodifikasi volume root, dan menambahkan Amazon EBS tambahan serta volume penyimpanan instans, sebagai berikut:

i. Volume root ditentukan dalam baris pertama.

- Untuk mengubah ukuran volume root, untuk Ukuran, masukkan nilai yang diperlukan.
- Jika Anda memilih Hapus saat pengakhiran, saat Anda mengakhiri instans yang dibuat dari AMI ini, volume EBS akan dihapus. Jika Anda mengosongkan Hapus saat pengakhiran, saat Anda mengakhiri instans, volume EBS tidak dihapus. Untuk informasi selengkapnya, lihat Pertahankan data saat instans diakhiri.

ii. Untuk menambahkan volume EBS, pilih Tambahkan Volume (yang akan menambahkan baris baru). Untuk Tipe Penyimpanan, pilih EBS, dan isi bidang dalam baris. Saat Anda meluncurkan instans dari AMI baru, volume tambahan ini secara otomatis dilampirkan ke instans. Volume kosong harus diformat dan dipasang. Volume berdasarkan snapshot harus dipasang.

iii. Untuk menambahkan volume penyimpanan instans, lihat Menambahkan volume penyimpanan instans ke AMI Amazon EC2. Saat Anda meluncurkan instans dari AMI baru, volume tambahan ini secara otomatis diinisialisasi dan dipasang. Volume ini tidak berisi data dari volume penyimpanan instans pada instans berjalan yang Anda gunakan untuk membuat AMI.

e. Tujuan snapshot — Jika volume instans Anda berada di Zona Lokal yang mendukung snapshot lokal EBS, pilih tempat untuk membuat snapshot AMI:

- Wilayah AWS: Buat snapshot di Wilayah induk Zona Lokal volume Anda.
- AWS Zona Lokal: Buat snapshot di Zona Lokal yang sama dengan volume Anda.

> **Note**
>
> Opsi ini hanya muncul di Zona Lokal yang mendukung snapshot lokal EBS, dan hanya jika instance Anda dibuat di Zona Lokal. Jika volume berada di Wilayah, opsi ini tidak muncul, dan snapshot secara otomatis dibuat di Wilayah yang sama dengan volume. Untuk informasi selengkapnya, lihat Snapshot lokal di Zona Lokal di Panduan Pengguna Amazon EBS.

> **Important**
>
> Semua snapshot volume instance harus berada di lokasi yang sama. Verifikasi lokasi snapshot yang ada. Jika ada snapshot yang ada di lokasi yang berbeda dari tujuan yang Anda pilih, pembuatan AMI akan gagal.

f. Tag – Anda dapat menandai AMI dan snapshot dengan tag yang sama, atau Anda dapat menandai keduanya dengan tag yang berbeda.

- Untuk menandai AMI dan snapshot dengan tag yang sama, pilih Tag gambar dan snapshot bersama-sama. Tag yang sama akan diterapkan pada AMI dan setiap snapshot yang dibuat.
- Untuk menandai AMI dan snapshot dengan tanda yang berbeda, pilih Tag gambar dan snapshot secara terpisah. Tag yang berbeda akan diterapkan pada AMI dan setiap snapshot yang dibuat. Namun, semua snapshot mendapatkan tag yang sama; Anda tidak dapat menandai setiap snapshot dengan tag yang berbeda.

Untuk menambahkan tag , pilih Tambahkan tag dan masukkan kunci dan nilai tag. Ulangi hal itu untuk setiap tanda.

g. Saat Anda siap membuat AMI, pilih Buat gambar.

5. Untuk melihat status AMI Anda saat sedang dibuat:

a. Di panel navigasi, pilih AMI.

b. Atur filter ke Dimiliki oleh saya, dan cari AMI Anda dalam daftar.

   Awalnya, statusnya adalah `pending` namun akan berubah menjadi `available` setelah beberapa menit.

6. (Opsional) Untuk melihat snapshot yang dibuat untuk AMI baru:

   a. Perhatikan ID AMI Anda di langkah sebelumnya.

   b. Di panel navigasi, pilih Snapshot.

   c. Atur filter ke Dimiliki oleh saya, lalu cari snapshot dengan ID AMI baru di kolom Deskripsi.

      Saat Anda meluncurkan instance dari AMI ini, Amazon EC2 menggunakan snapshot ini untuk membuat volume root instans.

AWS CLI

Untuk membuat AMI

Gunakan perintah create-image.

```
aws ec2 create-image \
    --instance-id i-1234567890abcdef0 \
    --name "my-web-server" \
    --description "My web server image" \
    --no-reboot
```

PowerShell

Untuk membuat AMI

Gunakan New-EC2Image cmdlet.

```
New-EC2Image `
    -InstanceId i-1234567890abcdef0 `
    -Name "my-web-server" `
    -Description "My web server image" `
    -NoReboot $true
```

## Buat AMI dari snapshot

Jika Anda memiliki snapshot dari volume root instance, Anda dapat membuat AMI dari snapshot ini.

> **Note**
>
> Dalam kebanyakan kasus, AMI untuk Windows, Red Hat, SUSE, dan SQL Server memerlukan informasi lisensi yang benar untuk hadir di AMI. Untuk informasi selengkapnya, lihat Memahami informasi penagihan AMI. Saat membuat AMI dari snapshot, `RegisterImage` operasi memperoleh informasi penagihan yang benar dari metadata snapshot, tetapi ini memerlukan metadata yang sesuai untuk hadir. Untuk memverifikasi apakah informasi penagihan yang benar diterapkan, periksa bidang Detail Platform pada AMI baru. Jika bidang kosong atau tidak cocok dengan kode sistem operasi yang diharapkan (misalnya, Windows, Red Hat, SUSE, atau SQL), pembuatan AMI tidak berhasil, dan Anda harus membuang AMI dan mengikuti instruksi diMembuat AMI dari instance.

Console

Untuk membuat AMI dari snapshot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Pilih snapshot untuk membuat AMI, lalu pilih Tindakan, Buat gambar dari snapshot.
4. Pada halaman Buat gambar dari snapshot, tentukan informasi berikut:
   a. Untuk Nama gambar, masukkan nama deskriptif untuk gambar tersebut.
   b. Untuk Deskripsi, masukkan deskripsi singkat untuk gambar tersebut.
   c. Untuk Arsitektur, pilih arsitektur gambar. Pilih i386 untuk 32-bit, x86_64 untuk 64-bit, arm64 untuk ARM 64-bit, atau x86_64 untuk macOS 64-bit.
   d. Untuk nama perangkat Root, masukkan nama perangkat yang akan digunakan untuk volume root. Untuk informasi selengkapnya, lihat Nama perangkat untuk volume di EC2 instans Amazon.
   e. Untuk Tipe virtualisasi, pilih tipe virtualisasi yang akan digunakan oleh instans yang diluncurkan dari AMI ini. Untuk informasi selengkapnya, lihat Tipe virtualisasi.
   f. (Hanya untuk virtualisasi paravirtual) Untuk ID Kernel, pilih kernel sistem operasi untuk gambar tersebut. Jika Anda menggunakan snapshot dari volume root instance, pilih

ID kernel yang sama dengan instance aslinya. Jika Anda tidak yakin, gunakan kernel default.

g. (Hanya untuk virtualisasi paravirtual) Untuk ID disk RAM, pilih disk RAM untuk gambar tersebut. Jika Anda memilih kernel tertentu, Anda mungkin perlu memilih RAM disk tertentu dengan driver yang mendukungnya.

h. Untuk mode Boot, pilih mode boot untuk gambar, atau pilih Gunakan default sehingga ketika sebuah instance diluncurkan dengan AMI ini, ia melakukan boot dengan mode boot yang didukung oleh jenis instance. Untuk informasi selengkapnya, lihat Mengatur mode boot dari AMI Amazon EC2.

i. (Opsional) Di bawah Blok pemetaan perangkat, sesuaikan volume root dan tambahkan volume data tambahan.

   Untuk setiap volume, Anda dapat menentukan ukuran, tipe, karakteristik performa, perilaku penghapusan saat pengakhiran, dan status enkripsi. Untuk volume root, ukurannya tidak bisa lebih kecil dari ukuran snapshot. Untuk tipe volume, General Purpose SSD gp3 adalah pilihan default.

j. (Opsional) Di bawah Tag, Anda dapat menambahkan satu atau lebih tag ke AMI baru. Untuk menambahkan tag , pilih Tambahkan tag dan masukkan kunci dan nilai tag. Ulangi hal itu untuk setiap tanda.

k. Saat Anda siap membuat AMI, pilih Buat gambar.

5. (Hanya Windows, Red Hat, SUSE, dan SQL Server) Untuk memverifikasi apakah informasi penagihan yang benar diterapkan, periksa bidang Detail Platform pada AMI baru. Jika bidang kosong atau tidak cocok dengan kode sistem operasi yang diharapkan (misalnya, Windows atau Red Hat), pembuatan AMI tidak berhasil, dan Anda harus membuang AMI dan mengikuti instruksi diMembuat AMI dari instance.

AWS CLI

Untuk membuat AMI dari snapshot menggunakan AWS CLI

Gunakan perintah register-image.

```
aws ec2 register-image \
    --name my-image \
    --root-device-name /dev/xvda \
    --block-device-mappings DeviceName=/dev/
xvda,Ebs={SnapshotId=snap-0db2cf683925d191f}
```

PowerShell

Untuk membuat AMI dari snapshot menggunakan PowerShell

Gunakan Register-EC2Image cmdlet.

```
$block = @{SnapshotId=snap-0db2cf683925d191f}
Register-EC2Image `
    -Name my-image `
    -RootDeviceName /dev/xvda `
    -BlockDeviceMapping @{DeviceName="/dev/xvda";Ebs=$block}
```

## Buat S3-backed AMI Amazon

AMI yang Anda tentukan saat meluncurkan instans menentukan jenis volume root.

Untuk membuat AMI Amazon S3-backed Linux, mulailah dari instance yang telah Anda luncurkan dari AMI Amazon S3-backed Linux yang sudah ada. Setelah Anda mengustomisasi instans sesuai kebutuhan, paketkan dan daftarkan AMI baru tersebut, yang dapat Anda gunakan untuk meluncurkan instans baru dengan kustomisasi ini.

Anda tidak dapat membuat AMI Amazon S3-backed Windows karena AMI Windows tidak mendukung penyimpanan instans untuk volume root.

> **Important**
>
> Hanya jenis instans berikut yang mendukung volume penyimpanan instans sebagai volume root dan memerlukan Amazon S3-backed AMI: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

Proses pembuatan AMI berbeda untuk EBS-backed AMI Amazon. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.

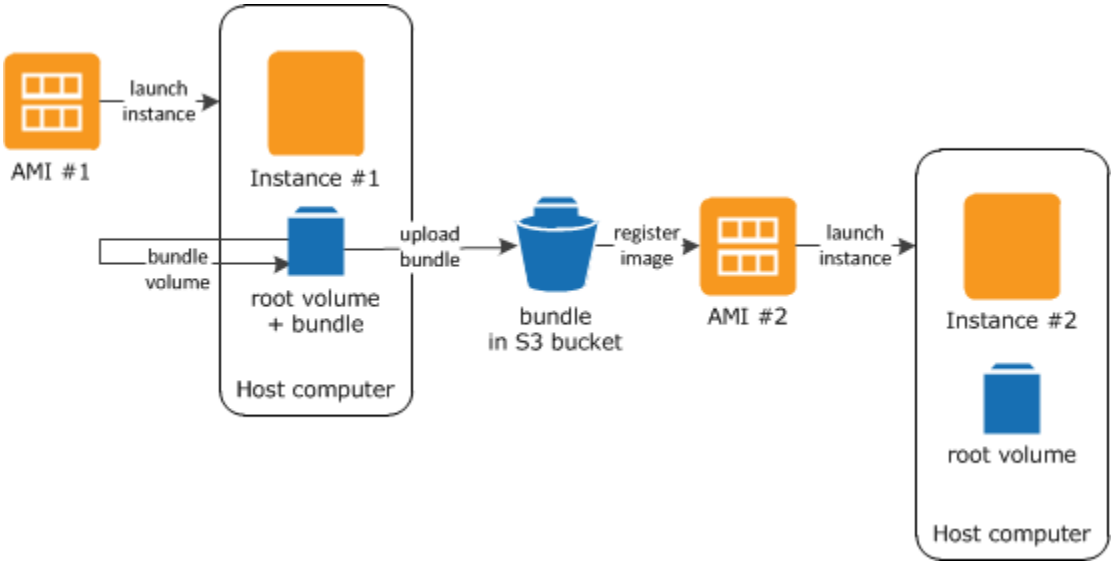
Daftar Isi


- Ikhtisar pembuatan AMI
- Prasyarat
- Membuat AMI dari instans Amazon Linux
- Menyiapkan alat Amazon EC2 AMI

- Referensi alat AMI Amazon EC2
- Konversikan Amazon S3-backed AMI Anda menjadi EBS-backed AMI

## Ikhtisar pembuatan AMI

Diagram berikut merangkum proses pembuatan AMI dari instance dengan volume root penyimpanan instance.




Pertama, luncurkan instans dari AMI yang serupa dengan AMI yang ingin Anda buat. Anda dapat menyambungkan ke instans dan mengustomisasikannya. Saat instans diatur sesuai keinginan, Anda dapat membuat paketan. Perlu beberapa menit untuk menyelesaikan proses pembuatan paket. Setelah proses selesai, Anda akan memiliki paketan, yang terdiri atas manifes gambar (`image.manifest.xml`) dan file (`image.part.xx`) yang berisi templat untuk volume root. Berikutnya, Anda mengunggah paketan tersebut ke bucket Amazon S3 kemudian mendaftarkan AMI Anda.

> **Note**
>
> Untuk mengunggah objek ke bucket S3 untuk AMI Amazon S3-backed Linux Anda, ACL harus diaktifkan untuk bucket. Jika tidak, Amazon EC2 tidak akan dapat mengatur ACL pada objek yang akan diunggah. Jika bucket tujuan Anda menggunakan pengaturan untuk Kepemilikan Objek S3 oleh pemilik bucket, hal ini tidak akan berfungsi karena ACL dinonaktifkan. Untuk informasi selengkapnya, lihat Mengontrol kepemilikan objek dan menonaktifkan ACL untuk bucket Anda.

Saat Anda meluncurkan instans menggunakan AMI yang baru, kami membuat volume root untuk instans tersebut menggunakan paketan yang Anda unggah ke Amazon S3. Ruang penyimpanan yang digunakan oleh paketan di Amazon S3 akan menimbulkan biaya sampai Anda menghapusnya. Untuk informasi selengkapnya, lihat Batalkan pendaftaran ke Amazon EC2 AMI.

Jika Anda menambahkan volume penyimpanan instans ke instans selain volume root, pemetaan perangkat blok untuk AMI baru berisi informasi untuk volume ini, dan pemetaan perangkat blok untuk instans yang Anda luncurkan dari AMI baru secara otomatis berisi informasi untuk volume ini. Untuk informasi selengkapnya, lihat Memblokir pemetaan perangkat untuk volume pada instans Amazon EC2.

## Prasyarat

Sebelum dapat membuat AMI, Anda harus menyelesaikan tugas berikut:

- Menginstal alat AMI. Untuk informasi selengkapnya, lihat Menyiapkan alat Amazon EC2 AMI.
- Instal AWS CLI. Untuk informasi selengkapnya, lihat Memulai dengan AWS CLI.
- Pastikan Anda memiliki bucket S3 untuk paketan, dan bucket Anda mengaktifkan ACL. Untuk informasi selengkapnya tentang mengonfigurasi ACL, lihat Mengonfigurasi ACL.
  - Untuk membuat bucket S3 menggunakan Konsol Manajemen AWS, buka konsol Amazon S3 di https://console.aws.amazon.com/s3/ dan pilih Buat Bucket.
  - Untuk membuat bucket S3 dengan AWS CLI, Anda dapat menggunakan perintah mb. Jika versi alat AMI yang terinstal adalah 1.5.18 atau yang lebih baru, Anda juga dapat menggunakan perintah `ec2-upload-bundle` untuk membuat bucket S3. Untuk informasi selengkapnya, lihat ec2-upload-bundle.
- Pastikan file dalam bundel Anda tidak dienkripsi di bucket S3. Jika Anda memerlukan enkripsi untuk AMI Anda, Anda dapat menggunakan EBS-backed AMI sebagai gantinya. Untuk informasi selengkapnya, lihat Gunakan enkripsi dengan EBS-backed AMI.
- Pastikan Anda memiliki ID AWS akun Anda. Untuk informasi selengkapnya, lihat Meli Akun AWS hat pen gidentifikasi di Panduan Referensi Manajemen AWS Akun.
- Pastikan Anda memiliki kredensial untuk menggunakan AWS CLI. Untuk informasi selengkapnya, lihat Otentikasi dan kredentif akses untuk AWS CLI di Panduan AWS Command Line Interface Pengguna.
- Pastikan Anda memiliki X.509 sertifikat dan kunci pribadi yang sesuai.
  - Jika Anda perlu membuat X.509 sertifikat, lihatMengelola sertifikat penandatanganan. X.509 Sertifikat dan kunci pribadi digunakan untuk mengenkripsi dan mendekripsi AMI Anda.

- [Tiongkok (Beijing)] Gunakan sertifikat `$EC2_AMITOOL_HOME/etc/ec2/amitools/cert-ec2-cn-north-1.pem`.
- [AWS GovCloud (US-West)] Gunakan `$EC2_AMITOOL_HOME/etc/ec2/amitools/cert-ec2-gov.pem` sertifikat.

- Sambungkan ke instans Anda dan kustomisasikan. Misalnya, Anda dapat menginstal perangkat lunak dan aplikasi, menyalin data, menghapus file sementara, dan mengubah konfigurasi Linux.

## Membuat AMI dari instans Amazon Linux

Prosedur berikut menjelaskan cara membuat AMI dari instance dengan volume root penyimpanan instance yang menjalankan Amazon Linux 1. Mereka mungkin tidak berfungsi untuk instance yang menjalankan distribusi Linux lainnya.

Untuk mempersiapkan penggunaan alat AMI (khusus instans HVM)

1. Alat AMI memerlukan GRUB Legacy untuk melakukan boot dengan benar. Gunakan perintah berikut untuk menginstal GRUB:

   ```
   [ec2-user ~]$ sudo yum install -y grub
   ```

2. Instal paket manajemen partisi dengan perintah berikut:

   ```
   [ec2-user ~]$ sudo yum install -y gdisk kpartx parted
   ```

Cara membuat AMI dari instans Amazon Linux dengan volume root penyimpanan instance

Prosedur ini mengasumsikan bahwa Anda telah memenuhi prasyarat dalam Prasyarat.

Dalam perintah berikut, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

1. Unggah kredensial Anda ke instans. Kami menggunakan kredensial ini untuk memastikan hanya Anda dan Amazon EC2 yang dapat mengakses AMI.

   a. Buat direktori sementara pada instans untuk kredensial Anda sebagai berikut:

      ```
      [ec2-user ~]$ mkdir /tmp/cert
      ```

Ini memungkinkan Anda mengecualikan kredensial Anda dari gambar yang dibuat.

b. Salin X.509 sertifikat dan kunci pribadi yang sesuai dari komputer Anda ke `/tmp/cert` direktori pada instance Anda menggunakan alat penyalinan aman seperti scp. `-i my-private-key.pem`Opsi dalam scp perintah berikut adalah kunci pribadi yang Anda gunakan untuk terhubung ke instance Anda dengan SSH, bukan kunci X.509 pribadi. Contoh:

```
you@your_computer:~ $ scp -i my-private-key.pem /
path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem /
path/to/cert-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem ec2-
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/
pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem  100%  717     0.7KB/s   00:00
cert-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem  100%  685     0.7KB/s   00:00
```

Atau, karena ini adalah file teks biasa, Anda dapat membuka sertifikat dan mengetikannya dalam editor teks dan menyalin kontennya ke dalam file baru di `/tmp/cert`.

2. Siapkan paketan untuk diunggah ke Amazon S3 dengan menjalankan perintah ec2-bundle-vol dari dalam instans Anda. Pastikan untuk menentukan opsi `-e` untuk mengecualikan direktori tempat kredensial Anda disimpan. Secara default, proses paketan mengecualikan file yang mungkin berisi informasi sensitif. File ini termasuk `*.sw`, `*.swo`, `*.swp`, `*.pem`, `*.priv`, `*id_rsa*`, `*id_dsa*` `*.gpg`, `*.jks`, `*/.ssh/authorized_keys`, dan `*/.bash_history`. Untuk menyertakan semua file ini, gunakan opsi `--no-filter`. Untuk menyertakan beberapa file ini, gunakan opsi `--include`.

> **Important**
>
> Secara default, proses pembuatan paketan AMI membuat koleksi file yang dikompresi dan dienkripsi di direktori `/tmp` yang mewakili volume root Anda. Jika tidak memiliki cukup ruang disk di `/tmp` untuk menyimpan paketan, Anda perlu menentukan lokasi berbeda untuk paketan yang akan disimpan dengan opsi `-d` */path/to/bundle/storage*. Beberapa instans memiliki penyimpanan sementara yang dipasang di `/mnt` atau `/media/ephemeral0` yang dapat Anda gunakan, atau Anda juga dapat membuat, melampirkan, dan memasang volume Amazon EBS) baru untuk menyimpan bundel. Untuk informasi selengkapnya, lihat Membuat volume Amazon EBS di Panduan Pengguna  Amazon EBS.

a. Anda harus menjalankan perintah ec2-bundle-vol sebagai root. Untuk sebagian besar perintah, Anda dapat menggunakan sudo untuk mendapatkan izin yang lebih tinggi, tetapi dalam kasus ini, Anda harus menjalankan sudo -E su untuk menjaga variabel lingkungan Anda.

```
[ec2-user ~]$ sudo -E su
```

Perhatikan bahwa perintah bash kini mengidentifikasi Anda sebagai pengguna root, dan bahwa tanda dolar telah diganti dengan tanda pagar, menandakan bahwa Anda berada dalam shell root:

```
[root ec2-user]#
```

b. Untuk membuat paketan AMI, jalankan perintah ec2-bundle-vol sebagai berikut:

```
[root ec2-user]# ec2-bundle-vol -k /tmp/cert/pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -c /tmp/cert/cert-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -u 123456789012 -r x86_64 -e /tmp/cert --
partition gpt
```

> **Note**
>
> Untuk Wilayah China (BeijingUS-West) dan AWS GovCloud (), gunakan `--ec2cert` parameter dan tentukan sertifikat sesuai prasyarat.

Perlu waktu beberapa menit untuk membuat gambar. Ketika perintah ini selesai, direktori Anda `/tmp` (atau non-default) berisi bundel (`image.manifest.xml`, ditambah beberapa `image.part.` *xx* file).

c. Keluar dari shell root.

```
[root ec2-user]# exit
```

3. (Opsional) Untuk menambahkan volume penyimpanan instans, edit pemetaan perangkat blok dalam file `image.manifest.xml` untuk AMI Anda. Untuk informasi selengkapnya, lihat Memblokir pemetaan perangkat untuk volume pada instans Amazon EC2.

a. Membuat cadangan file `image.manifest.xml` Anda.

```
[ec2-user ~]$ sudo cp /tmp/image.manifest.xml /tmp/image.manifest.xml.bak
```

b. Format ulang file `image.manifest.xml` agar lebih mudah dibaca dan diubah.

```
[ec2-user ~]$ sudo xmllint --format /tmp/image.manifest.xml.bak > /tmp/
image.manifest.xml
```

c. Edit pemetaan perangkat blok di `image.manifest.xml` editor teks. Contoh di bawah ini menunjukkan entri baru untuk volume penyimpanan instans `ephemeral1`.

> **Note**
>
> Untuk daftar file yang dikecualikan, lihat ec2-bundle-vol.

```
    <block_device_mapping>
      <mapping>
        <virtual>ami</virtual>
        <device>sda</device>
      </mapping>
      <mapping>
        <virtual>ephemeral0</virtual>
        <device>sdb</device>
      </mapping>
      <mapping>
        <virtual>ephemeral1</virtual>
        <device>sdc</device>
      </mapping>
      <mapping>
        <virtual>root</virtual>
        <device>/dev/sda1</device>
      </mapping>
    </block_device_mapping>
```

d. Simpan file `image.manifest.xml`, dan tutup editor teks Anda.

4. Untuk mengunggah paketan Anda ke Amazon S3, jalankan perintah ec2-upload-bundle sebagai berikut.

```
[ec2-user ~]$ ec2-upload-bundle -b amzn-s3-demo-bucket/bundle_folder/bundle_name -
m /tmp/image.manifest.xml -a your_access_key_id -s your_secret_access_key
```

> **Important**
>
> Untuk mendaftarkan AMI Anda di Wilayah selain AS Timur (Virginia Utara), Anda harus menentukan Wilayah target dengan opsi `--region` dan jalur bucket yang sudah ada di Wilayah target atau jalur bucket unik yang dapat dibuat di Wilayah target.

5. (Opsional) Setelah bundel diunggah ke Amazon S3, Anda dapat menghapus bundel dari direktori `/tmp` pada instans menggunakan perintah rm berikut:

```
[ec2-user ~]$ sudo rm /tmp/image.manifest.xml /tmp/image.part.* /tmp/image
```

> **Important**
>
> Jika Anda menentukan jalur dengan opsi `-d /path/to/bundle/storage` di Step 2, gunakan jalur tersebut, alih-alih `/tmp`.

6. Untuk mendaftarkan AMI Anda, jalankan perintah register-image sebagai berikut.

```
[ec2-user ~]$ aws ec2 register-image --image-location amzn-s3-demo-
bucket/bundle_folder/bundle_name/image.manifest.xml --name AMI_name --
virtualization-type hvm
```

> **Important**
>
> Jika sebelumnya Anda menentukan Wilayah untuk perintah ec2-upload-bundle, tentukan Wilayah itu untuk perintah ini.

## Menyiapkan alat Amazon EC2 AMI

Anda dapat menggunakan alat AMI untuk membuat dan mengelola AMI Amazon S3-backed Linux. Untuk menggunakan alat ini, Anda harus menginstalnya di instans Linux Anda. Alat AMI tersedia baik sebagai RPM dan sebagai file .zip untuk distribusi Linux yang tidak mendukung RPM.

Untuk menyiapkan alat AMI menggunakan RPM

1. Instal Ruby menggunakan manajer paket untuk distribusi Linux Anda, seperti yum. Sebagai contoh:

```
[ec2-user ~]$ sudo yum install -y ruby
```

2. Unduh file RPM menggunakan alat seperti wget atau curl. Sebagai contoh:

```
[ec2-user ~]$ wget https://s3.amazonaws.com/ec2-downloads/ec2-ami-tools.noarch.rpm
```

3. Verifikasikan tanda tangan file RPM menggunakan perintah berikut:

```
[ec2-user ~]$ rpm -K ec2-ami-tools.noarch.rpm
```

   Perintah di atas harus menunjukkan bahwa hash SHA1 dan MD5 file adalah `OK`. Jika perintah menunjukkan bahwa hash adalah `NOT OK`, gunakan perintah berikut untuk melihat hash Header SHA1 dan MD5 file:

```
[ec2-user ~]$ rpm -Kv ec2-ami-tools.noarch.rpm
```

   Kemudian, bandingkan hash Header SHA1 dan MD5 file Anda dengan alat AMI terverifikasi berikut ini untuk mengonfirmasi keaslian file:

   - Header SHA1: a1f662d6f25f69871104e6a62187fa4df508f880
   - MD5: 9faff05258064e2f7909b66142de6782

   Jika hash Header SHA1 dan MD5 file Anda cocok dengan hash alat AMI yang terverifiaksi, lanjutkan ke langkah berikutnya.

4. Instal RPM menggunakan perintah berikut:

```
[ec2-user ~]$ sudo yum install ec2-ami-tools.noarch.rpm
```

5. Verifikasi instalasi alat AMI Anda menggunakan perintah ec2-ami-tools-version.

```
[ec2-user ~]$ ec2-ami-tools-version
```

> **Note**
>
> Jika Anda menerima kesalahan pemuatan seperti “tidak dapat memuat file seperti itu -- ec2/amitools /version (LoadError)”, selesaikan langkah berikutnya untuk menambahkan lokasi instalasi alat AMI Anda ke RUBYLIB jalur Anda.

6. (Opsional) Jika Anda menerima kesalahan pada langkah sebelumnya, tambahkan lokasi instalasi alat AMI Anda ke jalur RUBYLIB.

   a. Jalankan perintah berikut untuk menentukan jalur yang akan ditambahkan.

   ```
   [ec2-user ~]$ rpm -qil ec2-ami-tools | grep ec2/amitools/version
   /usr/lib/ruby/site_ruby/ec2/amitools/version.rb
   /usr/lib64/ruby/site_ruby/ec2/amitools/version.rb
   ```

   Dalam contoh di atas, file yang hilang dari kesalahan pemuatan sebelumnya berada di /usr/lib/ruby/site_ruby dan /usr/lib64/ruby/site_ruby.

   b. Tambahkan lokasi dari langkah sebelumnya ke jalur RUBYLIB Anda.

   ```
   [ec2-user ~]$ export RUBYLIB=$RUBYLIB:/usr/lib/ruby/site_ruby:/usr/lib64/ruby/
   site_ruby
   ```

   c. Verifikasi instalasi alat AMI Anda menggunakan perintah ec2-ami-tools-version.

   ```
   [ec2-user ~]$ ec2-ami-tools-version
   ```

Untuk menyiapkan alat AMI menggunakan file .zip

1. Instal Ruby dan bongkar file .zip menggunakan pengelola paket untuk distribusi Linux Anda, seperti apt-get. Sebagai contoh:

   ```
   [ec2-user ~]$ sudo apt-get update -y && sudo apt-get install -y ruby unzip
   ```

2. Unduh file .zip menggunakan alat seperti wget atau curl. Sebagai contoh:

   ```
   [ec2-user ~]$ wget https://s3.amazonaws.com/ec2-downloads/ec2-ami-tools.zip
   ```

3. Bongkar file .zip ke dalam direktori instalasi yang sesuai, seperti /usr/local/ec2.

```
[ec2-user ~]$ sudo mkdir -p /usr/local/ec2
$ sudo unzip ec2-ami-tools.zip -d /usr/local/ec2
```

Perhatikan bahwa file.zip berisi folder ec2-ami-tools-. *x* *x*. *x*, dimana*x*. *x*. *x*adalah nomor versi alat (misalnya,`ec2-ami-tools-1.5.7`).

4. Atur variabel lingkungan `EC2_AMITOOL_HOME` untuk direktori instalasi alat. Sebagai contoh:

```
[ec2-user ~]$ export EC2_AMITOOL_HOME=/usr/local/ec2/ec2-ami-tools-x.x.x
```

5. Tambahkan alat ke variabel lingkungan `PATH` Anda. Sebagai contoh:

```
[ec2-user ~]$ export PATH=$EC2_AMITOOL_HOME/bin:$PATH
```

6. Anda dapat memverifikasi instalasi alat AMI Anda menggunakan perintah ec2-ami-tools-version.

```
[ec2-user ~]$ ec2-ami-tools-version
```

Mengelola sertifikat penandatanganan

Perintah tertentu dalam alat AMI memerlukan sertifikat penandatanganan (juga dikenal sebagai X.509 sertifikat). Anda harus membuat sertifikat dan kemudian mengunggahnya ke AWS. Misalnya, Anda dapat menggunakan alat pihak ketiga, seperti OpenSSL untuk membuat sertifikat.

Untuk membuat sertifikat tanda tangan

1. Instal dan konfigurasi OpenSSL.
2. Buat kunci privat menggunakan perintah `openssl genrsa` dan simpan output-nya ke file `.pem`. Kami sarankan Anda membuat kunci RSA 2048- atau 4096-bit.

```
openssl genrsa 2048 > private-key.pem
```

3. Munculkan sertifikat menggunakan perintah `openssl req`.

```
openssl req -new -x509 -nodes -sha256 -days 365 -key private-key.pem -outform PEM -
out certificate.pem
```

Untuk mengunggah sertifikat ke AWS, gunakan perintah upload-sig ning-certificate.

```
aws iam upload-signing-certificate --user-name user-name --certificate-body
 file://path/to/certificate.pem
```

Untuk mencantumkan sertifikat bagi pengguna, gunakan perintah list-signing-certificates:

```
aws iam list-signing-certificates --user-name user-name
```

Untuk menonaktifkan atau mengaktifkan kembali sertifikat penandatangan bagi pengguna, gunakan perintah update-signing-certificate. Perintah berikut ini menonaktifkan sertifikat:

```
aws iam update-signing-certificate --certificate-id OFHPLP4ZULTHYPMSYEX7O4BEXAMPLE --
status Inactive --user-name user-name
```

Untuk menghapus sertifikat, gunakan perintah delete-signing-certificate:

```
aws iam delete-signing-certificate --user-name user-name --certificate-
id OFHPLP4ZULTHYPMSYEX7O4BEXAMPLE
```

## Referensi alat AMI Amazon EC2

Anda dapat menggunakan perintah AMI tools untuk membuat dan mengelola AMI Amazon S3-backed Linux. Untuk menyiapkan alat, lihat Menyiapkan alat Amazon EC2 AMI.

Untuk informasi tentang kunci akses Anda, lihat M engelola kunci akses untuk pengguna IAM di Panduan Pengguna IAM.

Commands

- ec2-ami-tools-version
- ec2-bundle-image
- ec2-bundle-vol
- ec2-delete-bundle
- ec2-download-bundle
- ec2-migrate-manifest
- ec2-unbundle
- ec2-upload-bundle

- Opsi umum untuk alat AMI

ec2-ami-tools-version

Deskripsi

Menjelaskan versi alat AMI.

Sintaksis

**ec2-ami-tools-version**

Output

Informasi versi.

Contoh

Perintah contoh ini menampilkan informasi versi untuk alat AMI yang Anda gunakan.

```
[ec2-user ~]$ ec2-ami-tools-version
1.5.2 20071010
```

ec2-bundle-image

Deskripsi

Membuat AMI Amazon S3-backed Linux dari gambar sistem operasi yang dibuat dalam file loopback.

Sintaksis

**ec2-bundle-image -c *path* -k *path* -u *account* -i *path* [-d *path*] [--ec2cert *path*] [-r *architecture*] [--productcodes *code1*,*code2*,...] [-B *mapping*] [-p *prefix*]**

Opsi

jalur `-c, --cert`

File sertifikat kunci publik RSA yang dienkode PEM.

Wajib: Ya

jalur `-k, --privatekey`

Jalur ke file kunci PEM-encoded RSA. Anda akan perlu menentukan kunci ini untuk membuka paketan ini, jadi simpanlah di tempat yang aman. Perhatikan bahwa kunci tidak harus didaftarkan ke AWS akun Anda.

Wajib: Ya

akun `-u, --user`

ID AWS akun pengguna, tanpa tanda hubung.

Wajib: Ya

jalur `-i, --image`

Jalur ke gambar yang akan dipaketkan.

Wajib: Ya

jalur `-d, --destination`

Direktori untuk membuat paketan.

Default: `/tmp`

Wajib: Tidak

jalur `--ec2cert`

Jalur ke sertifikat kunci X.509 publik Amazon EC2 yang digunakan untuk mengenkripsi manifes gambar.

Wilayah `us-gov-west-1` dan `cn-north-1` menggunakan sertifikat kunci publik non-default dan jalur ke sertifikat tersebut harus ditetapkan dengan opsi ini. Jalur ke sertifikat berbeda-beda tergantung metode instalasi alat AMI. Untuk Amazon Linux, sertifikat terletak di `/opt/aws/amitools/ec2/etc/ec2/amitools/`. Jika Anda menginstal alat AMI dari file RPM atau ZIP di Menyiapkan alat Amazon EC2 AMI, sertifikat berada di `$EC2_AMITOOL_HOME/etc/ec2/amitools/`.

Wajib: Hanya untuk Wilayah `us-gov-west-1` dan `cn-north-1`.

arsitektur `-r, --arch`

Arsitektur gambar. Jika Anda tidak memberikan arsitektur pada baris perintah, Anda akan diminta memberikannya saat pembuatan paketan dimulai.

Nilai yang valid: `i386` | `x86_64`

Wajib: Tidak

kode1,kode2,... `--productcodes`

Kode produk yang dilampirkan ke gambar pada waktu pendaftaran, dipisahkan dengan koma.

Wajib: Tidak

pemetaan `-B, --block-device-mapping`

Menentukan bagaimana perangkat blok terpapar ke instans AMI ini jika tipe instans mendukung perangkat yang ditentukan.

Tentukan daftar pasangan kunci-nilai yang dipisahkan koma, di mana setiap kunci adalah nama virtual dan setiap nilai adalah nama perangkat yang terkait. Nama virtual mencakup hal berikut:

- `ami`—Perangkat sistem file root, seperti yang terlihat oleh instans
- `root`—Perangkat sistem file root, seperti yang terlihat oleh kernel
- `swap`—Perangkat pertukaran, seperti yang terlihat oleh instans
- `ephemeralN`—Volume penyimpanan instans ke-N

Wajib: Tidak

awalan `-p, --prefix`

Awalan nama file untuk file AMI yang dipaketkan.

Default: Nama file gambar. Misalnya, jika jalur gambar adalah `/var/spool/my-image/version-2/debian.img`, awalan default-nya adalah `debian.img`.

Wajib: Tidak

kernel_id `--kernel`

Tidak lagi digunakan. Gunakan register-image untuk mengatur kernel.

Wajib: Tidak

ramdisk_id `--ramdisk`

Tidak lagi digunakan. Gunakan register-image untuk mengatur disk RAM jika diperlukan.

Wajib: Tidak

Output

Pesan status yang menjelaskan tahap dan status proses pembuatan paketan.

Contoh

Contoh ini membuat paketan AMI dari gambar sistem operasi yang dibuat dalam file loopback.

```
[ec2-user ~]$ ec2-bundle-image -k pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -c cert-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -u 111122223333 -i image.img -d bundled/ -r x86_64
Please specify a value for arch [i386]:
Bundling image file...
Splitting bundled/image.gz.crypt...
Created image.part.00
Created image.part.01
Created image.part.02
Created image.part.03
Created image.part.04
Created image.part.05
Created image.part.06
Created image.part.07
Created image.part.08
Created image.part.09
Created image.part.10
Created image.part.11
Created image.part.12
Created image.part.13
Created image.part.14
Generating digests for each part...
Digests generated.
Creating bundle manifest...
ec2-bundle-image complete.
```

ec2-bundle-vol

Deskripsi

Membuat AMI Amazon S3-backed Linux dengan mengompresi, mengenkripsi, dan menandatangani salinan volume root untuk instance.

Amazon EC2 berupaya untuk mewarisi kode produk, pengaturan kernel, pengaturan disk RAM, dan pemetaan perangkat blok dari instans.

Secara default, proses paketan mengecualikan file yang mungkin berisi informasi sensitif. File ini termasuk `*.sw`, `*.swo`, `*.swp`, `*.pem`, `*.priv`, `*id_rsa*`, `*id_dsa*` `*.gpg`, `*.jks`, `*/.ssh/authorized_keys`, dan `*/.bash_history`. Untuk menyertakan semua file ini, gunakan opsi `--no-filter`. Untuk menyertakan beberapa file ini, gunakan opsi `--include`.

Untuk informasi selengkapnya, lihat Buat S3-backed AMI Amazon.

Sintaksis

```
ec2-bundle-vol -c path -k path -u account [-d path] [--ec2cert path] [-r architecture] [--productcodes code1,code2,...] [-B mapping] [--all] [-e directory1,directory2,...] [-i file1,file2,...] [--no-filter] [-p prefix] [-s size] [--[no-]inherit] [-v volume] [-P type] [-S script] [--fstab path] [--generate-fstab] [--grub-config path]
```

Opsi

jalur `-c, --cert`

File sertifikat kunci publik RSA yang dienkode PEM.

Wajib: Ya

jalur `-k, --privatekey`

Jalur ke file kunci PEM-encoded RSA pengguna.

Wajib: Ya

akun `-u, --user`

ID AWS akun pengguna, tanpa tanda hubung.

Wajib: Ya

tujuan `-d, --destination`

Direktori untuk membuat paketan.

Default: `/tmp`

Wajib: Tidak

jalur `--ec2cert`

Jalur ke sertifikat kunci X.509 publik Amazon EC2 yang digunakan untuk mengenkripsi manifes gambar.

Wilayah `us-gov-west-1` dan `cn-north-1` menggunakan sertifikat kunci publik non-default dan jalur ke sertifikat tersebut harus ditetapkan dengan opsi ini. Jalur ke sertifikat berbeda-beda tergantung metode instalasi alat AMI. Untuk Amazon Linux, sertifikat terletak di `/opt/aws/amitools/ec2/etc/ec2/amitools/`. Jika Anda menginstal alat AMI dari file RPM atau ZIP di Menyiapkan alat Amazon EC2 AMI, sertifikat berada di `$EC2_AMITOOL_HOME/etc/ec2/amitools/`.

Wajib: Hanya untuk Wilayah `us-gov-west-1` dan `cn-north-1`.

arsitektur `-r, --arch`

Arsitektur gambar. Jika Anda tidak memberikannya pada baris perintah, Anda akan diminta untuk memberikannya saat pembuatan paketan dimulai.

Nilai yang valid: `i386` | `x86_64`

Wajib: Tidak

kode1,kode2,... `--productcodes`

Kode produk yang dilampirkan ke gambar pada waktu pendaftaran, dipisahkan dengan koma.

Wajib: Tidak

pemetaan `-B, --block-device-mapping`

Menentukan bagaimana perangkat blok terpapar ke instans AMI ini jika tipe instans mendukung perangkat yang ditentukan.

Tentukan daftar pasangan kunci-nilai yang dipisahkan koma, di mana setiap kunci adalah nama virtual dan setiap nilai adalah nama perangkat yang terkait. Nama virtual mencakup hal berikut:

- `ami`—Perangkat sistem file root, seperti yang terlihat oleh instans
- `root`—Perangkat sistem file root, seperti yang terlihat oleh kernel
- `swap`—Perangkat pertukaran, seperti yang terlihat oleh instans
- `ephemeralN`—Volume penyimpanan instans ke-N

Wajib: Tidak

`-a, --all`

Membuat paketan semua direktori, termasuk yang ada di sistem file yang dipasang dari jauh.

Wajib: Tidak

direktori1,direktori2,... `-e, --exclude`

Daftar jalur dan file direktori mutlak yang akan dikecualikan dari operasi paketan. Parameter ini menimpa opsi `--all`. Jika ada pengecualian, direktori dan subdirektori yang tercantum dengan parameter tidak akan dipaketkan dengan volume.

Wajib: Tidak

file1,file2,... `-i, --include`

Daftar file yang akan disertakan dalam operasi paketan. File yang ditentukan akan dikecualikan dari AMI karena mungkin mengandung informasi sensitif.

Wajib: Tidak

`--no-filter`

Jika ditentukan, kami tidak akan mengecualikan file dari AMI karena file tersebut mungkin berisi informasi sensitif.

Wajib: Tidak

awalan `-p, --prefix`

Awalan nama file untuk file AMI yang dipaketkan.

Default: `image`

Wajib: Tidak

ukuran `-s, --size`

Ukuran, dalam MB (1024 * 1024 byte), dari file gambar yang akan dibuat. Ukuran maksimalnya adalah 10240 MB.

Default: 10240

Wajib: Tidak

`--[no-]inherit`

Mengindikasikan apakah gambar harus mewarisi metadata instans (default-nya adalah mewarisi). Pembuatan paketan gagal jika Anda mengaktifkan `--inherit`, tetapi metadata instans tidak dapat diakses.

Wajib: Tidak

volume `-v, --volume`

Jalur mutlak ke volume yang dipasang dari tempat untuk membuat paketan.

Default: Direktori root (/)

Wajib: Tidak

tipe `-P, --partition`

Menunjukkan apakah gambar disk harus menggunakan tabel partisi. Jika Anda tidak menentukan tipe tabel partisi, default-nya adalah tipe yang digunakan pada perangkat blok induk volume, jika berlaku. Jika tidak, default-nya adalah `gpt`.

Nilai yang valid: `mbr` | `gpt` | `none`

Wajib: Tidak

skrip `-S, --script`

Skrip kustomisasi akan dijalankan tepat sebelum pembuatan paketan. Skrip harus menantikan satu argumen, titik pemasangan volume.

Wajib: Tidak

jalur `--fstab`

Jalur ke fstab yang akan dipaketkan ke dalam gambar. Jika ini tidak ditentukan, Amazon EC2 bundel/etc/fstab.

Wajib: Tidak

`--generate-fstab`

Bundel volume menggunakan Amazon EC2-provided fstab.

Wajib: Tidak

`--grub-config`

Jalur menuju file konfigurasi grub alternatif untuk dipaketkan ke dalam gambar. Secara default, `ec2-bundle-vol` menunggu `/boot/grub/menu.lst` atau `/boot/grub/grub.conf` ada pada gambar hasil klona. Opsi ini memungkinkan Anda menentukan jalur ke file konfigurasi grub alternatif, yang kemudian akan disalin menggantikan default (jika ada).

Wajib: Tidak

kernel_id `--kernel`

Tidak lagi digunakan. Gunakan register-image untuk mengatur kernel.

Wajib: Tidak

ramdisk_id`--ramdisk`

Tidak lagi digunakan. Gunakan register-image untuk mengatur disk RAM jika diperlukan.

Wajib: Tidak

Output

Pesan status yang menjelaskan tahap dan status proses pemaketan.

Contoh

Contoh ini membuat AMI paketan dengan mengompresi, mengenkripsi, dan menandatangani snapshot sistem file root mesin lokal.

```
[ec2-user ~]$ ec2-bundle-vol -d /mnt -k pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -c
 cert-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -u 111122223333 -r x86_64
  Copying / into the image file /mnt/image...
  Excluding:
       sys
       dev/shm
       proc
       dev/pts
       proc/sys/fs/binfmt_misc
       dev
       media
       mnt
```

```
      proc
      sys
      tmp/image
      mnt/img-mnt
1+0 records in
1+0 records out
mke2fs 1.38 (30-Jun-2005)
warning: 256 blocks unused.

Splitting /mnt/image.gz.crypt...
Created image.part.00
Created image.part.01
Created image.part.02
Created image.part.03
...
Created image.part.22
Created image.part.23
Generating digests for each part...
Digests generated.
Creating bundle manifest...
Bundle Volume complete.
```

ec2-delete-bundle

Deskripsi

Menghapus paketan tertentu dari penyimpanan Amazon S3. Setelah menghapus paketan, Anda tidak dapat meluncurkan instans dari AMI terkait.

Sintaksis

**ec2-delete-bundle -b *bucket* -a *access_key_id* -s *secret_access_key* [-t *token*] [--url *url*] [--region *region*] [--sigv *version*] [-m *path*] [-p *prefix*] [--clear] [--retry] [-y]**

Opsi

bucket `-b, --bucket`

Nama bucket Amazon S3 yang berisi paketan AMI, diikuti dengan awalan jalur opsional yang dibatasi ‘/’

Wajib: Ya

access_key_id `-a, --access-key`

ID kunci AWS akses.

Wajib: Ya

secret_access_key `-s, --secret-key`

K AWS unci akses rahasia.

Wajib: Ya

token `-t, --delegation-token`

Token delegasi untuk diteruskan ke AWS permintaan. Untuk informasi selengkapnya, lihat Kredentif keamanan sementara di Panduan Pengguna IAM.

Wajib: Hanya saat Anda menggunakan kredensial keamanan sementara.

Default: Nilai dari variabel lingkungan `AWS_DELEGATION_TOKEN` (jika diatur).

wilayah`--region`

Wilayah yang akan digunakan dalam tanda tangan permintaan.

Default: `us-east-1`

Wajib: Wajib jika menggunakan tanda tangan versi 4

version `--sigv`

Versi tanda tangan yang digunakan ketika menandatangani permintaan.

Nilai yang valid: 2 | 4

Default: 4

Wajib: Tidak

path`-m, --manifest`

Jalur ke file manifes.

Wajib: Anda harus menentukan `--prefix` atau `--manifest`.

awalan `-p, --prefix`

Awalan nama file AMI yang dipaketkan. Berikan seluruh awalan. Misalnya, jika awalannya adalah image.img, gunakan `-p image.img` dan bukan `-p image`.

Wajib: Anda harus menentukan `--prefix` atau `--manifest`.

`--clear`

Menghapus bucket Amazon S3 jika kosong setelah menghapus paketan tertentu.

Wajib: Tidak

`--retry`

Otomatis mencoba ulang semua kesalahan Amazon S3, hingga lima kali per operasi.

Wajib: Tidak

`-y, --yes`

Secara otomatis mengasumsikan jawaban semua permintaan adalah ya.

Wajib: Tidak

Output

Amazon EC2 menampilkan pesan status yang menunjukkan tahap dan status proses penghapusan.

Contoh

Contoh ini menghapus paketan dari Amazon S3.

```
[ec2-user ~]$ ec2-delete-bundle -b amzn-s3-demo-bucket -a your_access_key_id -
s your_secret_access_key
Deleting files:
amzn-s3-demo-bucket/image.manifest.xml
amzn-s3-demo-bucket/image.part.00
amzn-s3-demo-bucket/image.part.01
amzn-s3-demo-bucket/image.part.02
amzn-s3-demo-bucket/image.part.03
amzn-s3-demo-bucket/image.part.04
amzn-s3-demo-bucket/image.part.05
```

```
amzn-s3-demo-bucket/image.part.06
Continue? [y/n]
y
Deleted amzn-s3-demo-bucket/image.manifest.xml
Deleted amzn-s3-demo-bucket/image.part.00
Deleted amzn-s3-demo-bucket/image.part.01
Deleted amzn-s3-demo-bucket/image.part.02
Deleted amzn-s3-demo-bucket/image.part.03
Deleted amzn-s3-demo-bucket/image.part.04
Deleted amzn-s3-demo-bucket/image.part.05
Deleted amzn-s3-demo-bucket/image.part.06
ec2-delete-bundle complete.
```

ec2-download-bundle

Deskripsi

Unduh AMI Amazon S3-backed Linux yang ditentukan dari penyimpanan Amazon S3.

Sintaksis

**ec2-download-bundle -b *bucket* -a *access_key_id* -s *secret_access_key* -k *path* [--url *url*] [--region *region*] [--sigv *version*] [-m *file*] [-p *prefix*] [-d *directory*] [--retry]**

Opsi

bucket `-b, --bucket`

Nama bucket Amazon S3 tempat paketan berada, diikuti dengan awalan jalur opsional yang dibatasi '/'.

Wajib: Ya

access_key_id `-a, --access-key`

ID kunci AWS akses.

Wajib: Ya

secret_access_key `-s, --secret-key`

K AWS unci akses rahasia.

Wajib: Ya

jalur `-k, --privatekey`

Kunci privat yang digunakan untuk mendekripsi manifes.

Wajib: Ya

url `--url`

URL layanan Amazon S3.

Default: `https://s3.amazonaws.com/`

Wajib: Tidak

wilayah `--region`

Wilayah yang akan digunakan dalam tanda tangan permintaan.

Default: `us-east-1`

Wajib: Wajib jika menggunakan tanda tangan versi 4

versi `--sigv`

Versi tanda tangan yang digunakan ketika menandatangani permintaan.

Nilai yang valid: `2` | `4`

Default: `4`

Wajib: Tidak

file `-m, --manifest`

Nama file manifes (tanpa jalur). Kami sarankan Anda untuk menentukan manifes (`-m`) atau awalan(`-p`).

Wajib: Tidak

awalan `-p, --prefix`

Awalan nama file untuk file AMI yang dipaketkan.

Default: `image`

Wajib: Tidak

direktori `-d, --directory`

Direktori tempat paketan yang diunduh disimpan. Direktori harus ada.

Default: Direktori kerja saat ini.

Wajib: Tidak

`--retry`

Otomatis mencoba ulang semua kesalahan Amazon S3, hingga lima kali per operasi.

Wajib: Tidak

Output

Pesan status yang menunjukkan berbagai tahap proses pengunduhan ditampilkan.

Contoh

Contoh ini membuat direktori `bundled` (menggunakan perintah Linux mkdir) dan mengunduh paketan dari bucket Amazon S3 `amzn-s3-demo-bucket`.

```
[ec2-user ~]$ mkdir bundled
[ec2-user ~]$ ec2-download-bundle -b amzn-s3-demo-bucket/bundles/bundle_name
 -m image.manifest.xml -a your_access_key_id -s your_secret_access_key -k pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -d mybundle
Downloading manifest image.manifest.xml from amzn-s3-demo-bucket to mybundle/
image.manifest.xml ...
Downloading part image.part.00 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.00 ...
Downloaded image.part.00 from amzn-s3-demo-bucket
Downloading part image.part.01 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.01 ...
Downloaded image.part.01 from amzn-s3-demo-bucket
Downloading part image.part.02 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.02 ...
Downloaded image.part.02 from amzn-s3-demo-bucket
Downloading part image.part.03 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.03 ...
```

```
Downloaded image.part.03 from amzn-s3-demo-bucket
Downloading part image.part.04 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.04 ...
Downloaded image.part.04 from amzn-s3-demo-bucket
Downloading part image.part.05 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.05 ...
Downloaded image.part.05 from amzn-s3-demo-bucket
Downloading part image.part.06 from amzn-s3-demo-bucket/bundles/bundle_name to
 mybundle/image.part.06 ...
Downloaded image.part.06 from amzn-s3-demo-bucket
```

ec2-migrate-manifest

Deskripsi

Memodifikasi AMI Amazon S3-backed Linux (misalnya, sertifikat, kernel, dan disk RAM) sehingga mendukung Wilayah yang berbeda.

Sintaksis

**ec2-migrate-manifest -c *path* -k *path* -m *path* {(-a *access_key_id* -s *secret_access_key* --region *region*) | (--no-mapping)} [--ec2cert *ec2_cert_path*] [--kernel *kernel-id*] [--ramdisk *ramdisk_id*]**

Opsi

jalur `-c, --cert`

File sertifikat kunci publik RSA yang dienkode PEM.

Wajib: Ya

jalur `-k, --privatekey`

Jalur ke file kunci PEM-encoded RSA pengguna.

Wajib: Ya

jalur `--manifest`

Jalur ke file manifes.

Wajib: Ya

access_key_id `-a, --access-key`

ID kunci AWS akses.

Wajib: Wajib jika menggunakan pemetaan otomatis.

secret_access_key `-s, --secret-key`

K AWS unci akses rahasia.

Wajib: Wajib jika menggunakan pemetaan otomatis.

wilayah `--region`

Wilayah untuk dicari di file pemetaan.

Wajib: Wajib jika menggunakan pemetaan otomatis.

`--no-mapping`

Menonaktifkan pemetaan otomatis kernel dan disk RAM.

Selama migrasi, Amazon EC2 menggantikan kernel dan disk RAM dalam file manifes dengan kernel dan disk RAM yang dirancang untuk wilayah tujuan. Kecuali parameter `--no-mapping` diberikan, `ec2-migrate-bundle` dapat menggunakan operasi `DescribeRegions` dan `DescribeImages` untuk melakukan pemetaan otomatis.

Wajib: Wajib jika Anda tidak menyediakan opsi `-a`, `-s`, dan `--region` yang digunakan untuk pemetaan otomatis.

jalur `--ec2cert`

Jalur ke sertifikat kunci X.509 publik Amazon EC2 yang digunakan untuk mengenkripsi manifes gambar.

Wilayah `us-gov-west-1` dan `cn-north-1` menggunakan sertifikat kunci publik non-default dan jalur ke sertifikat tersebut harus ditetapkan dengan opsi ini. Jalur ke sertifikat berbeda-beda tergantung metode instalasi alat AMI. Untuk Amazon Linux, sertifikat terletak di `/opt/aws/amitools/ec2/etc/ec2/amitools/`. Jika Anda menginstal alat AMI dari file ZIP di Menyiapkan alat Amazon EC2 AMI, sertifikat berada di `$EC2_AMITOOL_HOME/etc/ec2/amitools/`.

Wajib: Hanya untuk Wilayah `us-gov-west-1` dan `cn-north-1`.

kernel_id `--kernel`

ID kernel yang akan dipilih.

> **Important**
>
> Kami menyarankan Anda menggunakan al PV-GRUB ih-alih kernel dan disk RAM. Untuk informasi selengkapnya, lihat Kernel yang disediakan pengguna di Panduan Pengguna Amazon Linux 2.

Wajib: Tidak

ramdisk_id `--ramdisk`

ID disk RAM yang akan dipilih.

> **Important**
>
> Kami menyarankan Anda menggunakan al PV-GRUB ih-alih kernel dan disk RAM. Untuk informasi selengkapnya, lihat Kernel yang disediakan pengguna di Panduan Pengguna Amazon Linux 2.

Wajib: Tidak

Output

Pesan status yang menjelaskan tahap dan status proses pembuatan paketan.

Contoh

Contoh ini menyalin AMI yang ditentukan dalam manifes `my-ami.manifest.xml` dari AS ke UE.

```
[ec2-user ~]$ ec2-migrate-manifest --manifest my-ami.manifest.xml
 --cert cert-HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem --privatekey pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBZQ55CLO.pem --region eu-west-1

Backing up manifest...
Successfully migrated my-ami.manifest.xml It is now suitable for use in eu-west-1.
```

ec2-unbundle

Deskripsi

Re-creates bundel dari AMI Amazon S3-backed Linux.

Sintaksis

**ec2-unbundle -k *path* -m *path* [-s *source_directory*] [-d *destination_directory*]**

Opsi

jalur `-k, --privatekey`

Jalur ke file kunci PEM-encoded RSA Anda.

Wajib: Ya

jalur `-m, --manifest`

Jalur ke file manifes.

Wajib: Ya

source_directory `-s, --source`

Direktori yang berisi paketan.

Default: Direktori saat ini.

Wajib: Tidak

destination_directory `-d, --destination`

Direktori tempat membongkar paketan AMI. Direktori tujuan harus ada.

Default: Direktori saat ini.

Wajib: Tidak

Contoh

Contoh Linux dan UNIX ini membongkar paketan AMI yang ditentukan dalam file `image.manifest.xml`.

```
[ec2-user ~]$ mkdir unbundled
$ ec2-unbundle -m mybundle/image.manifest.xml -k pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -s mybundle -d unbundled
$ ls -l unbundled
total 1025008
-rw-r--r-- 1 root root 1048578048 Aug 25 23:46 image.img
```

Output

Pesan status yang menunjukkan berbagai tahap proses pembongkaran paketan ditampilkan.

ec2-upload-bundle

Deskripsi

Mengunggah bundel untuk AMI Amazon S3-backed Linux ke Amazon S3 dan menetapkan daftar kontrol akses (ACL) yang sesuai pada objek yang diunggah. Untuk informasi selengkapnya, lihat Buat S3-backed AMI Amazon.

> **Note**
>
> Untuk mengunggah objek ke bucket S3 untuk AMI Amazon S3-backed Linux Anda, ACL harus diaktifkan untuk bucket. Jika tidak, Amazon EC2 tidak akan dapat mengatur ACL pada objek yang akan diunggah. Jika bucket tujuan Anda menggunakan pengaturan untuk Kepemilikan Objek S3 oleh pemilik bucket, hal ini tidak akan berfungsi karena ACL dinonaktifkan. Untuk informasi selengkapnya, lihat Mengontrol kepemilikan objek dan menonaktifkan ACL untuk bucket Anda.

Sintaksis

```
ec2-upload-bundle -b bucket -a access_key_id -s secret_access_key [-t
token] -m path [--url url] [--region region] [--sigv version] [--acl acl]
[-d directory] [--part part] [--retry] [--skipmanifest]
```

Opsi

bucket `-b, --bucket`

Nama bucket Amazon S3 untuk menyimpan paketan, diikuti dengan awalan jalur opsional yang dibatasi '/'. Jika bucket tidak tersedia, bucket akan dibuat jika namanya tersedia. Selain itu, jika

bucket tidak ada dan versi alat AMI adalah 1.5.18 atau yang lebih baru, perintah ini menetapkan ACL untuk bucket.

Wajib: Ya

access_key_id `-a, --access-key`

ID kunci AWS akses Anda.

Wajib: Ya

secret_access_key `-s, --secret-key`

Kunci akses AWS rahasiamu.

Wajib: Ya

token `-t, --delegation-token`

Token delegasi untuk diteruskan ke AWS permintaan. Untuk informasi selengkapnya, lihat Kredentif keamanan sementara di Panduan Pengguna IAM.

Wajib: Hanya saat Anda menggunakan kredensial keamanan sementara.

Default: Nilai dari variabel lingkungan `AWS_DELEGATION_TOKEN` (jika diatur).

jalur `-m, --manifest`

Jalur ke file manifes. File manifes dibuat selama proses pembuatan paketan dan dapat ditemukan di direktori yang berisi paketan tersebut.

Wajib: Ya

url `--url`

Tidak lagi digunakan. Gunakan opsi `--region` sebagai gantinya kecuali jika bucket Anda dibatasi ke lokasi `EU` (dan bukan `eu-west-1`). Bendera `--location` adalah satu-satunya cara untuk menarget batasan lokasi tertentu.

URL layanan titik akhir Amazon S3.

Default: `https://s3.amazonaws.com/`

Wajib: Tidak

wilayah `--region`

Wilayah yang akan digunakan dalam tanda tangan permintaan untuk bucket S3 tujuan.

- Jika bucket tidak ada dan Anda tidak menentukan Wilayah, alat akan membuat bucket tanpa batasan lokasi (di `us-east-1`).
- Jika bucket tidak ada dan Anda menentukan Wilayah, alat akan membuat bucket di Wilayah yang ditentukan.
- Jika bucket tersedia dan Anda tidak menentukan Wilayah, alat ini akan menggunakan lokasi bucket.
- Jika bucket tersedia dan Anda menentukan `us-east-1` sebagai Wilayah, alat ini akan menggunakan lokasi aktual bucket tanpa pesan kesalahan apa pun, setiap file yang cocok akan ditimpa.
- Jika bucket tersedia dan Anda menetapkan Wilayah (selain `us-east-1`) yang tidak sesuai dengan lokasi aktual bucket, alat akan keluar dengan kesalahan.

Jika bucket Anda dibatasi ke lokasi `EU` (dan bukan `eu-west-1`), gunakan bendera `--location` sebagai gantinya. Bendera `--location` adalah satu-satunya cara untuk menarget batasan lokasi tertentu.

Default: `us-east-1`

Wajib: Wajib jika menggunakan tanda tangan versi 4

versi `--sigv`

Versi tanda tangan yang digunakan ketika menandatangani permintaan.

Nilai yang valid: `2` | `4`

Default: `4`

Wajib: Tidak

acl `--acl`

Kebijakan daftar kontrol akses dari gambar yang dipaketkan.

Nilai yang valid: `public-read` | `aws-exec-read`

Default: `aws-exec-read`

Wajib: Tidak

direktori `-d, --directory`

Direktori yang berisi bagian AMI yang dipaketkan.

Default: Direktori yang berisi file manifes (lihat opsi `-m`).

Wajib: Tidak

bagian `--part`

Mulai mengunggah bagian tertentu dan semua bagian berikutnya. Sebagai contoh, `--part 04`.

Wajib: Tidak

`--retry`

Otomatis mencoba ulang semua kesalahan Amazon S3, hingga lima kali per operasi.

Wajib: Tidak

`--skipmanifest`

Tidak mengunggah manifes.

Wajib: Tidak

lokasi `--location`

Tidak lagi digunakan. Gunakan opsi `--region` sebagai gantinya, kecuali jika bucket Anda dibatasi ke lokasi `EU` (dan bukan `eu-west-1`). Bendera `--location` adalah satu-satunya cara untuk menarget batasan lokasi tersebut.

Batasan lokasi tujuan bucket Amazon S3. Jika bucket tersedia dan Anda menetapkan lokasi yang tidak sesuai dengan lokasi aktual bucket, alat akan keluar dengan kesalahan. Jika bucket tersedia dan Anda tidak menentukan lokasi, alat ini akan menggunakan lokasi bucket. Jika bucket tidak tersedia dan Anda menentukan lokasi, alat akan membuat bucket di lokasi yang ditentukan. Jika bucket tidak ada dan Anda tidak menentukan lokasi, alat akan membuat bucket tanpa batasan lokasi (di `us-east-1`).

Default: Jika `--region` ditentukan, lokasi diatur ke Wilayah yang ditentukan. Jika `--region` tidak ditentukan, lokasi secara default menjadi `us-east-1`.

Wajib: Tidak

Output

Amazon EC2 menampilkan pesan status yang menunjukkan tahap dan status proses pengunggahan.

Contoh

Contoh ini mengunggah paketan yang ditentukan oleh manifes `image.manifest.xml`.

```
[ec2-user ~]$ ec2-upload-bundle -b amzn-s3-demo-bucket/bundles/bundle_name -m
 image.manifest.xml -a your_access_key_id -s your_secret_access_key
Creating bucket...
Uploading bundled image parts to the S3 bucket amzn-s3-demo-bucket ...
Uploaded image.part.00
Uploaded image.part.01
Uploaded image.part.02
Uploaded image.part.03
Uploaded image.part.04
Uploaded image.part.05
Uploaded image.part.06
Uploaded image.part.07
Uploaded image.part.08
Uploaded image.part.09
Uploaded image.part.10
Uploaded image.part.11
Uploaded image.part.12
Uploaded image.part.13
Uploaded image.part.14
Uploading manifest ...
Uploaded manifest.
Bundle upload completed.
```

Opsi umum untuk alat AMI

Sebagian besar alat AMI menerima parameter opsional berikut.

`--help, -h`

Menampilkan pesan bantuan.

`--version`

Menampilkan versi dan pemberitahuan hak cipta.

`--manual`

Menampilkan entri manual.

`--batch`

Berjalan dalam mode batch, menahan perintah interaktif.

`--debug`

Menampilkan informasi yang dapat berguna saat pemecahan masalah.

## Konversikan Amazon S3-backed AMI Anda menjadi EBS-backed AMI

Anda dapat mengonversi AMI Amazon S3-backed Linux yang Anda miliki menjadi AMI Amazon EBS-backed Linux.

> **Important**
>
> Anda tidak dapat mengonversi AMI yang tidak Anda miliki.

Untuk mengonversi S3-backed AMI Amazon ke EBS-backed AMI Amazon

1. Luncurkan instance Amazon Linux dari Amazon EBS-backed AMI. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol. Instans Amazon Linux memiliki alat AWS CLI dan AMI yang sudah diinstal sebelumnya.
2. Unggah kunci X.509 pribadi yang Anda gunakan untuk menggabungkan Amazon S3-backed AMI ke instans Anda. Kami menggunakan kunci ini untuk memastikan hanya Anda dan Amazon EC2 yang dapat mengakses AMI.

   a. Buat direktori sementara pada instance Anda untuk kunci X.509 pribadi Anda sebagai berikut:

   ```
   [ec2-user ~]$ mkdir /tmp/cert
   ```

   b. Salin kunci X.509 pribadi Anda dari komputer Anda ke `/tmp/cert` direktori pada instance Anda, menggunakan alat penyalinan aman seperti scp. *my-private-key*Parameter dalam perintah berikut adalah kunci pribadi yang Anda gunakan untuk terhubung ke instance Anda dengan SSH. Contoh:

```
you@your_computer:~ $ scp -i my-private-key.pem /
path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem ec2-
user@ec2-203-0-113-25.compute-1.amazonaws.com:/tmp/cert/
pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem  100%  717     0.7KB/s   00:00
```

3. Konfigurasikan variabel lingkungan Anda untuk menggunakan AWS CLI. Untuk informasi selengkapnya, lihat Variabel lingkungan.

   a. (Disarankan) Tetapkan variabel lingkungan untuk kunci AWS akses, kunci rahasia, dan token sesi Anda.

   ```
   [ec2-user ~]$ export AWS_ACCESS_KEY_ID=your_access_key_id
   [ec2-user ~]$ export AWS_SECRET_ACCESS_KEY=your_secret_access_key
   [ec2-user ~]$ export AWS_SESSION_TOKEN=your_session_token
   ```

   b. Tetapkan variabel lingkungan untuk kunci AWS akses Anda, dan kunci rahasia.

   ```
   [ec2-user ~]$ export AWS_ACCESS_KEY_ID=your_access_key_id
   [ec2-user ~]$ export AWS_SECRET_ACCESS_KEY=your_secret_access_key
   ```

4. Siapkan volume Amazon Elastic Block Store (Amazon EBS) untuk AMI baru Anda.

   a. Buat volume EBS kosong di Zona Ketersediaan yang sama dengan instans Anda menggunakan perintah create-volume. Perhatikan ID volume di output perintah.

   > **Important**
   >
   > Volume EBS ini harus berukuran sama atau lebih besar dari volume root penyimpanan instans awal.

   ```
   aws ec2 create-volume \
       --size 10 \
       --region us-west-2 \
       --availability-zone us-west-2b
   ```

   b. Lampirkan volume ke EBS-backed instance Amazon Anda menggunakan perintah attach-volume.

   ```
   aws ec2 attach-volume \
   ```

```
    --volume-id vol-01234567890abcdef \
    --instance-id i-1234567890abcdef0 \
    --region us-west-2
```

5. Buat folder untuk paketan Anda.

```
[ec2-user ~]$ mkdir /tmp/bundle
```

6. Unduh paketan untuk AMI berbasis penyimpanan instans Anda ke `/tmp/bundle` menggunakan perintah ec2-download-bundle.

```
[ec2-user ~]$ ec2-download-bundle -b amzn-s3-demo-bucket/bundle_folder/bundle_name
 -m image.manifest.xml -a $AWS_ACCESS_KEY_ID -s $AWS_SECRET_ACCESS_KEY --
privatekey /path/to/pk-HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem -d /tmp/bundle
```

7. Susun kembali file gambar dari paketan menggunakan perintah ec2-unbundle.

   a. Ubah direktori ke folder paketan.

```
[ec2-user ~]$ cd /tmp/bundle/
```

   b. Jalankan perintah ec2-unbundle.

```
[ec2-user bundle]$ ec2-unbundle -m image.manifest.xml --privatekey /path/to/pk-
HKZYKTAIG2ECMXYIBH3HXV4ZBEXAMPLE.pem
```

8. Salin file dari gambar yang tidak dipaketkan ke volume EBS baru.

```
[ec2-user bundle]$ sudo dd if=/tmp/bundle/image of=/dev/sdb bs=1M
```

9. Periksa volume apakah ada partisi baru yang tidak dipaketkan.

```
[ec2-user bundle]$ sudo partprobe /dev/sdb1
```

10. Buat daftar perangkat blok untuk mencari nama perangkat yang akan dipasang.

```
[ec2-user bundle]$ lsblk
NAME         MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
/dev/sda    202:0    0   8G  0 disk
##/dev/sda1 202:1    0   8G  0 part /
/dev/sdb    202:80   0  10G  0 disk
##/dev/sdb1 202:81   0  10G  0 part
```

Dalam contoh ini, partisi yang akan dipasang adalah `/dev/sdb1`, tetapi nama perangkat Anda mungkin akan berbeda. Jika volume Anda tidak dipartisi, perangkat yang akan dipasang akan serupa dengan `/dev/sdb` (tanpa digit di bagian akhir partisi perangkat).

11. Buat titik pemasangan untuk volume EBS yang baru dan pasang volume.

```
[ec2-user bundle]$ sudo mkdir /mnt/ebs
[ec2-user bundle]$ sudo mount /dev/sdb1 /mnt/ebs
```

12. Buka file `/etc/fstab` pada volume EBS dengan editor teks favorit Anda (seperti vim atau nano) dan hapus entri apa pun untuk volume penyimpanan instans (ephemeral). Karena volume EBS dipasang di `/mnt/ebs`, file `fstab` berada di `/mnt/ebs/etc/fstab`.

```
[ec2-user bundle]$ sudo nano /mnt/ebs/etc/fstab
#
LABEL=/     /           ext4    defaults,noatime  1   1
tmpfs       /dev/shm    tmpfs   defaults        0   0
devpts      /dev/pts    devpts  gid=5,mode=620  0   0
sysfs       /sys        sysfs   defaults        0   0
proc        /proc       proc    defaults        0   0
/dev/sdb        /media/ephemeral0       auto    defaults,comment=cloudconfig    0
     2
```

Dalam contoh ini, baris terakhir harus dihapus.

13. Lepas volume dan pisahkan dari instans.

```
[ec2-user bundle]$ sudo umount /mnt/ebs
[ec2-user bundle]$ aws ec2 detach-volume --volume-id vol-01234567890abcdef --
region us-west-2
```

14. Buat AMI dari volume EBS baru sebagai berikut.

    a. Buat snapshot volume EBS yang baru.

```
[ec2-user bundle]$ aws ec2 create-snapshot --region us-west-2 --description
 "your_snapshot_description" --volume-id vol-01234567890abcdef
```

    b. Periksa untuk memastikan kelengkapan snapshot Anda.

```
[ec2-user bundle]$ aws ec2 describe-snapshots --region us-west-2 --snapshot-
id snap-0abcdef1234567890
```

c. Identifikasi arsitektur prosesor, tipe virtualisasi, dan gambar kernel (`aki`) yang digunakan pada AMI asli dengan perintah describe-images. Anda memerlukan ID AMI dari S3-backed AMI Amazon asli untuk langkah ini.

```
[ec2-user bundle]$ aws ec2 describe-images --region us-west-2 --image-
id ami-0abcdef1234567890 --output text
IMAGES x86_64 amazon/amzn-ami-pv-2013.09.2.x86_64-s3 ami-8ef297be amazon
 available public machine aki-fc8f11cc instance-store paravirtual xen
```

Dalam contoh ini, arsitekturnya adalah `x86_64` dan ID gambar kernel-nya adalah `aki-fc8f11cc`. Gunakan nilai-nilai ini di langkah berikut. Jika output perintah di atas juga mencantumkan ID `ari`, perhatikan juga hal tersebut.

d. Daftarkan AMI baru Anda dengan ID snapshot volume EBS baru Anda dan nilai-nilai dari langkah sebelumnya. Jika output perintah sebelumnya mencantumkan ID `ari`, sertakan ID tersebut dalam perintah berikut ini dengan `--ramdisk-id` *`ari_id`*.

```
[ec2-user bundle]$ aws ec2 register-image --region us-west-2 --
name your_new_ami_name --block-device-mappings DeviceName=device-
name,Ebs={SnapshotId=snap-0abcdef1234567890} --virtualization-type paravirtual
 --architecture x86_64 --kernel-id aki-fc8f11cc --root-device-name device-name
```

15. (Opsional) Setelah Anda menguji bahwa Anda dapat meluncurkan instans dari AMI baru, Anda dapat menghapus volume EBS yang Anda buat untuk prosedur ini.

```
aws ec2 delete-volume --volume-id vol-01234567890abcdef
```

## Buat AMI Amazon EC2 menggunakan Windows Sysprep

Alat Microsoft System Preparation (Windows Sysprep) membuat versi umum dari sistem operasi, dengan konfigurasi sistem khusus instance dihapus sebelum menangkap gambar baru.

Kami sarankan Anda menggunakan EC2 Image Builder untuk mengotomatiskan pembuatan, pengelolaan, dan deployment gambar server “golden” kustom yang aman dan terbaru serta telah terinstal dan terkonfigurasi dengan perangkat lunak dan pengaturan.

Anda juga dapat menggunakan Windows Sysprep untuk membuat AMI standar menggunakan agen peluncuran Windows: EC2launch v2, EC2launch, dan EC2config.

> **Important**
>
> Jangan gunakan Windows Sysprep untuk membuat cadangan instance. Windows Sysprep menghapus informasi khusus sistem; menghapus informasi ini mungkin memiliki konsekuensi yang tidak diinginkan untuk cadangan instance.

Untuk memecahkan masalah Windows Sysprep, lihat. Memecahkan masalah Sysprep dengan instans Amazon EC2 Windows

Daftar Isi


- Fase Windows Sysprep
- Sebelum Anda mulai
- Buat AMI menggunakan Windows Sysprep dengan EC2launch v2
- Buat AMI menggunakan Windows Sysprep dengan EC2launch
- Buat AMI menggunakan Windows Sysprep dengan EC2config


## Fase Windows Sysprep

Windows Sysprep berjalan melalui fase berikut:

- Generalisasi: Alat Sysprep menghapus informasi dan konfigurasi khusus gambar. Misalnya, Windows Sysprep menghapus pengenal keamanan (SID), nama komputer, log peristiwa, dan driver tertentu, untuk menyebutkan beberapa. Setelah fase ini selesai, sistem operasi (OS) siap untuk membuat AMI.

  > **Note**
  >
  > Ketika Anda menjalankan Windows Sysprep dengan agen peluncuran Windows, sistem mencegah driver dihapus karena `PersistAllDeviceInstalls` diatur ke true secara default.

- Spesialisasi: Plug and Play memindai komputer dan menginstal driver untuk perangkat yang terdeteksi. Alat Sysprep menghasilkan persyaratan OS, seperti nama komputer dan SID. Anda juga dapat menjalankan perintah dalam fase ini.
- Out-of-Box Pengalaman (OOBE): Sistem menjalankan versi singkat dari Windows Setup dan meminta Anda untuk memasukkan informasi seperti bahasa sistem, zona waktu, dan organisasi terdaftar. Saat Anda menjalankan Windows Sysprep dengan agen peluncuran Windows, file jawaban mengotomatiskan fase ini.

## Sebelum Anda mulai

- Sebelum melakukan Windows Sysprep, kami sarankan Anda menghapus semua akun pengguna lokal dan semua profil akun selain akun administrator tunggal di mana Windows Sysprep akan dijalankan. Jika Anda menjalankan Windows Sysprep dengan akun dan profil tambahan, perilaku tak terduga dapat terjadi, termasuk kehilangan data profil atau kegagalan untuk menyelesaikan Windows Sysprep.
- Pelajari lebih lanjut tentang Ikh tisar Sysprep.
- Pelajari Dukungan Sysprep untuk Peran Server.

## Buat AMI menggunakan Windows Sysprep dengan EC2launch v2

Saat Anda membuat gambar dari instance dengan agen EC2launch v2 diinstal, EC2launch v2 melakukan tugas tertentu saat gambar disiapkan. Ini termasuk bekerja dengan Windows Sysprep. Untuk informasi selengkapnya, lihat the section called “Fase Windows Sysprep”.

Daftar Isi


- Tindakan Windows Sysprep
- Pasca Sysprep
- Jalankan Windows Sysprep dengan EC2launch v2


### Tindakan Windows Sysprep

Windows Sysprep dan EC2launch v2 melakukan tindakan berikut saat menyiapkan gambar.

1. Saat Anda memilih Matikan dengan Sysprep di kotak dialog Pengaturan EC2Launch, sistem menjalankan perintah `ec2launch sysprep`.

2. EC2Launch v2 mengubah konten file `unattend.xml` dengan membaca nilai registri di `HKEY_USERS\.DEFAULT\Control Panel\International\LocaleName`. File ini terletak di direktori berikut: `C:\ProgramData\Amazon\EC2Launch\sysprep`.
3. Sistem menjalankan `BeforeSysprep.cmd`. Perintah ini membuat kunci registri sebagai berikut:

   reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 1 /f

   Kunci registri akan menonaktifkan koneksi RDP hingga diaktifkan kembali. Menonaktifkan koneksi RDP adalah langkah keamanan yang diperlukan karena, selama sesi boot pertama setelah Windows Sysprep berjalan, ada periode waktu singkat di mana RDP memungkinkan koneksi dan kata sandi Administrator kosong.
4. Layanan EC2Launch v2 memanggil Windows Sysprep dengan menjalankan perintah berikut:

   sysprep.exe /oobe /generalize /shutdown /unattend: "C:\ProgramData\Amazon\EC2Launch \sysprep\unattend.xml"

Fase Generalisasi

- EC2Launch v2 menghapus informasi dan konfigurasi spesifik gambar, seperti nama komputer dan SID. Jika instans adalah anggota sebuah domain, instans akan dihapus dari domain tersebut. File jawaban `unattend.xml` mencakup pengaturan berikut yang memengaruhi fase ini:
  - PersistAllDeviceInstalls: Pengaturan ini mencegah Pengaturan Windows menghapus dan mengonfigurasi ulang perangkat, yang mempercepat proses persiapan gambar karena AMI Amazon memerlukan driver tertentu untuk dijalankan dan deteksi ulang driver tersebut akan memakan waktu.
  - DoNotCleanUpNonPresentDevices: Pengaturan ini menyimpan informasi Plug and Play untuk perangkat yang saat ini tidak ada.
- Windows Sysprep mematikan OS saat bersiap untuk membuat AMI. Sistem meluncurkan instans baru atau memulai instans asal.

Tahap Spesialisasi

Sistem menghasilkan OS-specific persyaratan, seperti nama komputer dan SID. Sistem juga melakukan tindakan berikut berdasarkan konfigurasi yang Anda tentukan dalam file jawaban `unattend.xml`.

- CopyProfile: Windows Sysprep dapat dikonfigurasi untuk menghapus semua profil pengguna, termasuk profil Administrator bawaan. Pengaturan ini mempertahankan akun Administrator bawaan sehingga kustomisasi apa pun yang Anda buat pada akun tersebut dipindahkan ke gambar baru. Nilai bawaannya adalah `True`.

  CopyProfile mengganti profil default dengan profil administrator lokal yang ada. Semua akun yang Anda masuk setelah menjalankan Windows Sysprep menerima salinan profil itu dan isinya saat login pertama.

  Jika Anda tidak memiliki kustomisasi profil pengguna tertentu yang ingin Anda tampilkan ke gambar baru tersebut, ubah pengaturan ini menjadi `False`. Windows Sysprep akan menghapus semua profil pengguna (ini menghemat waktu dan ruang disk).
- TimeZone: Zona waktu diatur ke Coordinate Universal Time (UTC) secara default.
- Perintah sinkron dengan order 1: Sistem menjalankan perintah berikut, yang mengaktifkan akun administrator dan menentukan persyaratan kata sandi:

  ```
  net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /
  PASSWORDREQ:YES
  ```
- Perintah sinkron dengan order 2: Sistem mengacak kata sandi administrator. Langkah keamanan ini dirancang untuk mencegah instans dapat diakses setelah Windows Sysprep selesai jika Anda tidak mengkonfigurasi tugas. `setAdminAccount`

  Sistem menjalankan perintah berikut dari direktori agen peluncuran lokal Anda (`C:\Program Files\Amazon\EC2Launch\`).

  ```
  EC2Launch.exe internal randomize-password --username Administrator
  ```
- Untuk mengaktifkan koneksi desktop jarak jauh, sistem menetapkan kunci `fDenyTSConnections` registri Terminal Server ke false.

Fase OOBE

1. Sistem menentukan konfigurasi berikut menggunakan file jawaban EC2Launch v2:
   - `<InputLocale>en-US</InputLocale>`
   - `<SystemLocale>en-US</SystemLocale>`
   - `<UILanguage>en-US</UILanguage>`

- `<UserLocale>en-US</UserLocale>`
- `<HideEULAPage>true</HideEULAPage>`
- `<HideWirelessSetupInOOBE>true</HideWirelessSetupInOOBE>`
- `<ProtectYourPC>3</ProtectYourPC>`
- `<BluetoothTaskbarIconEnabled>false</BluetoothTaskbarIconEnabled>`
- `<TimeZone>UTC</TimeZone>`
- `<RegisteredOrganization>Amazon.com</RegisteredOrganization>`
- `<RegisteredOwner>EC2</RegisteredOwner>`

> **Note**
>
> Selama fase generalisasi dan spesialisasi, EC2Launch v2 memantau status OS. Jika EC2Launch v2 mendeteksi bahwa OS berada dalam fase Sysprep, ia kemudian menerbitkan pesan berikut ke log sistem:
> Windows sedang dikonfigurasi. SysprepState=IMAGE_STATE_TIDAK DAPAT DITERAPKAN

2. Sistem menjalankan EC2Launch v2.

Pasca Sysprep

Setelah Windows Sysprep selesai, EC2launch v2 mengirimkan pesan berikut ke output konsol:

```
Windows sysprep configuration complete.
```

EC2Launch v2 kemudian melakukan tindakan berikut:

1. Membaca konten file `agent-config.yml` dan menjalankan tugas-tugas yang dikonfigurasi.
2. Menjalankan semua tugas dalam tahap `preReady`.
3. Setelah selesai, mengirim pesan `Windows is ready` ke log sistem instans.
4. Menjalankan semua tugas dalam tahap `PostReady`.

Untuk informasi selengkapnya tentang EC2Launch v2, lihat Gunakan agen EC2launch v2 untuk melakukan tugas selama peluncuran instance EC2 Windows.

### Jalankan Windows Sysprep dengan EC2launch v2

Gunakan prosedur berikut untuk membuat AMI standar menggunakan Windows Sysprep dengan EC2launch v2.

1. Di konsol Amazon EC2, cari AMI yang ingin Anda duplikat.
2. Jalankan dan hubungkan ke instans Windows Anda.
3. Sesuaikan pengaturan
   a. Dari menu Start Windows, cari dan pilih pengaturan Amazon EC2Launch. Untuk informasi selengkapnya tentang opsi dan pengaturan di kotak dialog pengaturan EC2Launch Amazon, lihat Konfigurasikan pengaturan EC2launch v2 untuk instans Windows.
   b. Jika Anda telah membuat perubahan, pilih Simpan sebelum Anda mematikan.
4. Pilih Shutdown dengan Sysprep atau Shutdown tanpa Sysprep.

Ketika Anda diminta untuk mengonfirmasi bahwa Anda ingin menjalankan Windows Sysprep dan mematikan instance, pilih Ya. EC2launch v2 menjalankan Windows Sysprep. Selanjutnya, Anda keluar dari instans, dan instans tersebut padam. Jika Anda memeriksa halaman Instans di konsol Amazon EC2, status instan berubah dari `Running` ke `Stopping`, lalu ke `Stopped`. Ketika itu, aman untuk membuat AMI dari instans ini.

Anda dapat secara manual memanggil alat Windows Sysprep dari baris perintah menggunakan perintah berikut:

```
"%programfiles%\amazon\ec2launch\ec2launch.exe" sysprep --shutdown=true
```

## Buat AMI menggunakan Windows Sysprep dengan EC2launch

Saat Anda membuat gambar dari instance dengan agen EC2launch diinstal, EC2launch melakukan tugas tertentu saat gambar disiapkan. Ini termasuk bekerja dengan Windows Sysprep. Untuk informasi selengkapnya, lihat the section called “Fase Windows Sysprep”.

EC2launch menawarkan file jawaban default dan file batch untuk Windows Sysprep yang mengotomatiskan dan mengamankan proses persiapan gambar pada AMI Anda. Memodifikasi file ini bersifat opsional. File ini terletak di direktori berikut secara default: `C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep`.

> **Important**
>
> Jangan gunakan Windows Sysprep untuk membuat cadangan instance. Windows Sysprep menghapus informasi khusus sistem. Jika Anda menghapus informasi ini, mungkin ada konsekuensi yang tidak diinginkan untuk pencadangan instans.

Daftar Isi


- EC2launch jawaban dan file batch untuk Windows Sysprep
- Jalankan Windows Sysprep dengan EC2launch
- Perbarui metadata/KMS rute untuk Server 2016 dan yang lebih baru saat meluncurkan AMI khusus


## EC2launch jawaban dan file batch untuk Windows Sysprep

File jawaban EC2Launch dan file batch untuk Windows Sysprep meliputi:

`Unattend.xml`

Ini adalah file jawaban default. Jika Anda menjalankan `SysprepInstance.ps1` atau memilih ShutdownWithSysprep dalam antarmuka pengguna, sistem membaca pengaturan dari file ini.

`BeforeSysprep.cmd`

Sesuaikan file batch ini untuk menjalankan perintah sebelum EC2launch menjalankan Windows Sysprep.

`SysprepSpecialize.cmd`

Sesuaikan file batch ini untuk menjalankan perintah selama fase spesialisasi Windows Sysprep.

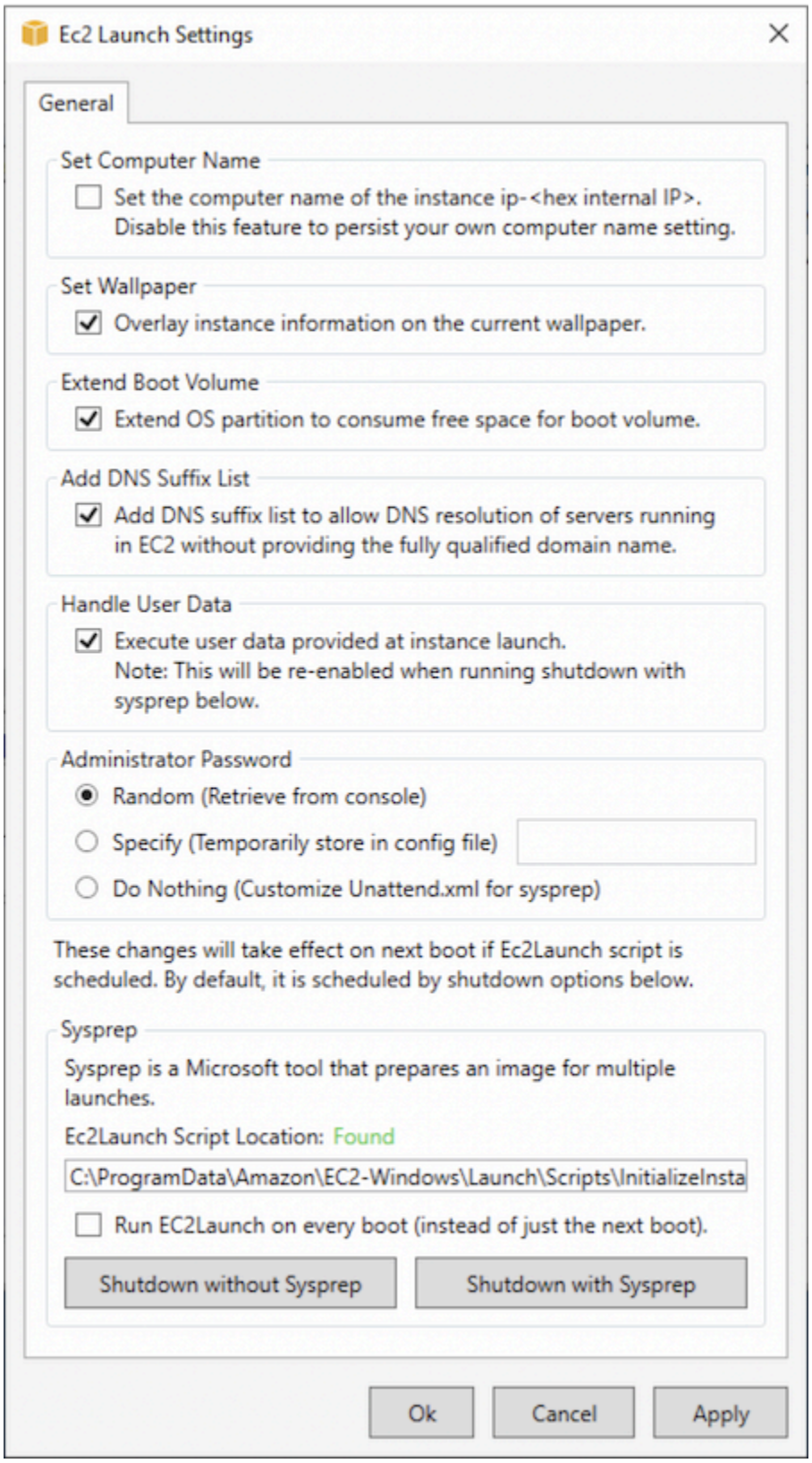
## Jalankan Windows Sysprep dengan EC2launch

Pada instalasi penuh Windows Server 2016 dan yang lebih baru (dengan pengalaman desktop), Anda dapat menjalankan Windows Sysprep dengan EC2launch secara manual atau dengan menggunakan aplikasi EC2 Launch Settings.

Untuk menjalankan Windows Sysprep menggunakan aplikasi Pengaturan EC2Launch

1. Di konsol Amazon EC2, cari atau buat AMI Windows Server 2016 atau yang lebih baru.

2. Luncurkan instans Windows dari AMI.
3. Sambungkan ke instans Windows Anda dan kustomisasikan.
4. Cari dan jalankan aplikasi EC2LaunchSettings. Aplikasi ini terletak di direktori berikut secara default: `C:\ProgramData\Amazon\EC2-Windows\Launch\Settings`.




5. Pilih atau hapus opsi sesuai kebutuhan. Pengaturan ini disimpan dalam file `LaunchConfig.json` Anda.

6. Untuk Kata Sandi Administrator, lakukan salah satu hal berikut:

   - Pilih Acak. EC2Launch membuat dan mengenkripsikan kata sandi menggunakan kunci pengguna. Sistem akan menonaktifkan pengaturan ini setelah instans dijalankan sehingga kata sandi akan tetap ada meskipun instans di-boot ulang atau dihentikan dan dimulai.
   - Pilih Tentukan dan ketik kata sandi yang memenuhi persyaratan sistem. Kata sandi disimpan `LaunchConfig.json` sebagai teks yang jelas dan dihapus setelah Windows Sysprep menetapkan kata sandi administrator. Jika Anda mematikan sekarang, kata sandi akan segera ditetapkan. EC2Launch mengenkripsi kata sandi menggunakan kunci pengguna.
   - Pilih DoNothing dan tentukan kata sandi dalam file `unattend.xml`. Jika Anda tidak menentukan kata sandi di `unattend.xml`, akun administrator akan dinonaktifkan.

7. Pilih Matikan dengan Sysprep.

Untuk menjalankan Windows Sysprep secara manual menggunakan EC2launch

1. Di konsol Amazon EC2, cari atau buat AMI edisi Datacenter Windows Server 2016 atau lebih baru yang ingin Anda duplikasi.
2. Luncurkan dan sambungkan ke instans Windows Anda.
3. Kustomisasikan instans.
4. Tentukan pengaturan di file `LaunchConfig.json`. File ini terletak di direktori `C:\ProgramData\Amazon\EC2-Windows\Launch\Config` secara default.

   Untuk `adminPasswordType`, tentukan satu dari nilai-nilai berikut:

   `Random`

   EC2Launch menmbuat dan mengenkripsikan kata sandi menggunakan kunci pengguna. Sistem akan menonaktifkan pengaturan ini setelah instans dijalankan sehingga kata sandi ini akan tetap ada jika instans tersebut di-boot ulang atau dihentikan dan dimulai.

   `Specify`

   EC2Launch menggunakan kata sandi yang Anda tentukan di `adminPassword`. Jika kata sandi tidak memenuhi persyaratan sistem, maka EC2Lauch membuat kata sandi acak sebagai gantinya. Kata sandi disimpan `LaunchConfig.json` sebagai teks yang jelas dan dihapus setelah Windows Sysprep menetapkan kata sandi administrator. EC2Launch mengenkripsi kata sandi menggunakan kunci pengguna.

`DoNothing`

EC2Launch menggunakan kata sandi yang Anda tentukan di file `unattend.xml`. Jika Anda tidak menentukan kata sandi di `unattend.xml`, akun administrator akan dinonaktifkan.

5. (Opsional) Tentukan pengaturan di `unattend.xml` dan file konfigurasi lainnya. Jika berencana mengatur instalasi, Anda tidak perlu mengubah file-file ini. File ini terletak di direktori berikut secara default: `C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep`.
6. Di Windows PowerShell, jalankan `./InitializeInstance.ps1 -Schedule`. Skrip ini terletak di direktori berikut secara default: `C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts`. Skrip ini menjadwalkan instans untuk diinisialisasi saat boot berikutnya. Anda harus menjalankan skrip ini sebelum menjalankan skrip `SysprepInstance.ps1` dalam langkah berikutnya.
7. Di Windows PowerShell, jalankan `./SysprepInstance.ps1`. Skrip ini terletak di direktori berikut secara default: `C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts`.

Anda keluar dari instans dan instans akan dimatikan. Jika Anda memeriksa halaman Instans di konsol Amazon EC2, status instan berubah dari `Running` ke `Stopping`, lalu ke `Stopped`. Saat itu, aman untuk membuat AMI dari instans ini.

Perbarui metadata/KMS rute untuk Server 2016 dan yang lebih baru saat meluncurkan AMI khusus

Untuk memperbarui metadata/KMS rute untuk Server 2016 dan yang lebih baru saat meluncurkan AMI khusus, lakukan salah satu hal berikut:

- Jalan EC2LaunchSettings kan GUI (C:ProgramData\\ Amazon\EC2-Windows\ Launch\ Sett Ec2LaunchSettings.exe ings\) dan pilih opsi untuk mematikan dengan Windows Sysprep.
- Jalan EC2LaunchSettings kan dan matikan tanpa Windows Sysprep sebelum membuat AMI. Ini akan mengatur tugas inisialisasi EC2 Launch untuk berjalan pada boot berikutnya, yang akan mengatur rute berdasarkan subnet untuk instans.
- Menjadwalkan ulang secara manual tugas inisialisasi EC2 Launch sebelum membuat AMI dari PowerShell.

> **Important**
>
> Perhatikan perilaku reset kata sandi default sebelum menjadwalkan ulang tugas.

- Untuk memperbarui rute pada instans yang mengalami aktivasi atau komunikasi Windows dengan kegagalan metadata instans, lihat “Tidak dapat mengaktivasi Windows”.

## Buat AMI menggunakan Windows Sysprep dengan EC2config

Saat Anda membuat gambar dari instance dengan layanan EC2config diinstal, EC2config melakukan tugas tertentu saat gambar disiapkan. Ini termasuk bekerja dengan Windows Sysprep. Untuk informasi selengkapnya, lihat the section called “Fase Windows Sysprep”.

Daftar Isi


- Tindakan Windows Sysprep
- Pasca Sysprep
- Jalankan Windows Sysprep dengan layanan EC2config


### Tindakan Windows Sysprep

Windows Sysprep dan layanan EC2config melakukan tindakan berikut saat menyiapkan gambar.

1. Saat Anda memilih Matikan dengan Sysprep di kotak dialog Properti Layanan EC2, sistem menjalankan perintah ec2config.exe –sysprep.
2. Layanan EC2Config membaca konten dari file `BundleConfig.xml`. Secara default, file terletak dalam direktori berikut: `C:\Program Files\Amazon\Ec2ConfigService\Settings`.

   File `BundleConfig.xml` mencakup pengaturan berikut. Anda dapat mengubah pengaturan ini:

   - AutoSysprep: Menunjukkan apakah akan menggunakan Windows Sysprep secara otomatis. Anda tidak perlu mengubah nilai ini jika Anda menjalankan Windows Sysprep dari kotak dialog Properti Layanan EC2. Nilai default-nya adalah `No`.
   - SetRDPCertificate: Mengatur sertifikat yang ditandatangani sendiri untuk server Desktop Jarak Jauh. Ini memungkinkan Anda menggunakan Remote Desktop Protocol (RDP) secara aman untuk tersambung ke instans. Ubah nilai ke `Yes` jika instans baru harus menggunakan sertifikat. Pengaturan ini tidak digunakan dengan instans Windows Server 2012 karena sistem operasi ini dapat menghasilkan sertifikat mereka sendiri. Nilai default-nya adalah `No`.
   - SetPasswordAfterSysprep: Menetapkan kata sandi acak pada instance yang baru diluncurkan, mengenkripsinya dengan kunci peluncuran pengguna, dan mengeluarkan kata sandi terenkripsi ke konsol. Ubah nilai ke `No` jika instans baru tidak boleh diatur ke kata sandi terenkripsi acak. Nilai default-nya adalah `Yes`.

- PreSysprepRunCmd: Lokasi perintah yang akan dijalankan. Skrip ini terletak di direktori berikut secara default: `C:\Program Files\Amazon\Ec2ConfigService\Scripts\BeforeSysprep.cmd`

3. Sistem menjalankan `BeforeSysprep.cmd`. Perintah ini membuat kunci registri sebagai berikut:

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v
 fDenyTSConnections /t REG_DWORD /d 1 /f
```

   Kunci registri akan menonaktifkan koneksi RDP hingga diaktifkan kembali. Menonaktifkan koneksi RDP adalah langkah keamanan yang diperlukan karena, selama sesi boot pertama setelah Windows Sysprep berjalan, ada periode waktu singkat di mana RDP memungkinkan koneksi dan kata sandi Administrator kosong.

4. Layanan EC2Config memanggil Windows Sysprep dengan menjalankan perintah berikut:

```
sysprep.exe /unattend: "C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml" /
oobe /generalize /shutdown
```

Fase Generalisasi

- Alat ini menghapus informasi dan konfigurasi spesifik gambar, seperti nama komputer dan SID. Jika instans adalah anggota sebuah domain, instans akan dihapus dari domain tersebut. File jawaban `sysprep2008.xml` mencakup pengaturan berikut yang memengaruhi fase ini:
  - PersistAllDeviceInstalls: Pengaturan ini mencegah Pengaturan Windows menghapus dan mengonfigurasi ulang perangkat, yang mempercepat proses persiapan gambar karena AMI Amazon memerlukan driver tertentu untuk dijalankan dan deteksi ulang driver tersebut akan memakan waktu.
  - DoNotCleanUpNonPresentDevices: Pengaturan ini menyimpan informasi Plug and Play untuk perangkat yang saat ini tidak ada.
- Windows Sysprep mematikan OS saat bersiap untuk membuat AMI. Sistem meluncurkan instans baru atau memulai instans asal.

Tahap Spesialisasi

Sistem menghasilkan persyaratan spesifik OS, seperti nama komputer dan SID. Sistem juga melakukan tindakan berikut ini berdasarkan konfigurasi yang Anda tentukan dalam file jawaban sysprep2008.xml.

- CopyProfile: Windows Sysprep dapat dikonfigurasi untuk menghapus semua profil pengguna, termasuk profil Administrator bawaan. Pengaturan ini mempertahankan akun Administrator bawaan sehingga kustomisasi apa pun yang Anda buat pada akun tersebut dipindahkan ke gambar baru. Nilai default-nya adalah True.

  CopyProfile mengganti profil default dengan profil administrator lokal yang ada. Semua akun yang masuk setelah menjalankan Windows Sysprep akan menerima salinan profil itu dan isinya pada login pertama.

  Apabila Anda tidak memiliki kustomisasi profil pengguna tertentu yang ingin Anda tampilkan ke gambar baru tersebut, ubah pengaturan ini menjadi False. Windows Sysprep akan menghapus semua profil pengguna; ini menghemat waktu dan ruang disk.
- TimeZone: Zona waktu diatur ke Coordinate Universal Time (UTC) secara default.
- Perintah sinkron dengan order 1: Sistem menjalankan perintah berikut, yang mengaktifkan akun administrator dan menentukan persyaratan kata sandi.

  net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER / PASSWORDREQ:YES
- Perintah sinkron dengan order 2: Sistem mengacak kata sandi administrator. Langkah keamanan ini dirancang untuk mencegah instans dapat diakses setelah Windows Sysprep selesai jika Anda tidak mengaktifkan setelan ec2setpassword.

  C:\Program File\ Amazon\Ec2ConfigService\ ScramblePassword.exe "-u Administrator
- Perintah sinkron dengan order 3: Sistem menjalankan perintah berikut:

  C:\Program File\ Amazon\Ec2ConfigService\ Skrip\ SysprepSpecializePhase.cmd

  Perintah ini menambahkan kunci registri berikut ini, yang mengaktifkan ulang RDP:

  reg tambahkan “HKEY_LOCAL_MACHINE\ SYSTEM\\ KontrolCurrentControlSet\ Terminal Server” /v fDenytsConnections /t REG_DWORD /d 0 /f

Fase OOBE

1. Menggunakan file jawaban layanan EC2Config, sistem menentukan konfigurasi berikut ini:
   - < InputLocale >en-US InputLocale </ >
   - < SystemLocale >en-US SystemLocale </ >
   - <UILanguage>en-US</UILanguage>
   - < UserLocale >en-US UserLocale </ >
   - <HideEULAPage>true</HideEULAPage>
   - < HideWirelessSetupInOOBE >benar</ HideWirelessSetupInOOBE >
   - < NetworkLocation >Lainnya</ NetworkLocation >
   - < ProtectYour PC> 3</PC> ProtectYour
   - < BluetoothTaskbarIconEnabled >salah </ BluetoothTaskbarIconEnabled >
   - < TimeZone >UTC TimeZone </>
   - <RegisteredOrganization>Amazon.com</RegisteredOrganization>
   - < RegisteredOwner >Amazon RegisteredOwner </>

   > **Note**
   >
   > Selama fase generalisasi dan spesialisasi, layanan EC2Config memantau status OS. Jika EC2Config mendeteksi bahwa OS berada dalam fase Sysprep, ia kemudian menerbitkan pesan berikut ke log sistem:
   > EC2ConfigMonitorState: 0 Windows sedang dikonfigurasi.
   > SysprepState=IMAGE_STATE_UNDEPLOYABLE

2. Setelah fase OOBE selesai, sistem menjalankan `SetupComplete.cmd` dari lokasi berikut: `C:\Windows\Setup\Scripts\SetupComplete.cmd`. Di AMI publik Amazon sebelum April 2015, file ini kosong dan tidak menjalankan apa pun pada gambar. Di AMI publik tertanggal setelah April 2015, file mencakup nilai berikut: call "C:\Program Files\Amazon\Ec2ConfigService\Scripts\PostSysprep.cmd".
3. Sistem menjalankan `PostSysprep.cmd`, yang melakukan operasi berikut:
   - Mengatur kata sandi Administrator lokal agar tidak kedaluwarsa. Jika kata sandi kedaluwarsa, Administrator mungkin tidak bisa masuk.
   - Mengatur nama mesin MSSQLServer (jika terinstal) sehingga nama akan tersinkron dengan AMI.

Pasca Sysprep

Setelah Windows Sysprep selesai, layanan EC2Config mengirimkan pesan berikut ke output konsol:

```
Windows sysprep configuration complete.
   Message: Sysprep Start
   Message: Sysprep End
```

EC2Config kemudian akan melakukan tindakan berikut:

1. Membaca konten file config.xml dan mencantumkan semua plug-in yang diaktifkan.
2. Menjalankan semua plug-in “Sebelum Windows siap” secara bersamaan.
   - Ec2SetPassword
   - Ec2SetComputerName
   - Ec2InitializeDrives
   - Ec2EventLog
   - Ec2ConfigureRDP
   - Ec2OutputRDPCert
   - Ec2SetDriveLetter
   - Ec2WindowsActivate
   - Ec2DynamicBootVolumeSize
3. Setelah selesai, mengirimkan pesan “Windows siap” ke log sistem instans.
4. Menjalankan semua plug-in “Setelah Windows siap” secara bersamaan.
   - CloudWatch Log Amazon
   - UserData
   - AWS Systems Manager (Manajer Sistem)

Untuk informasi selengkapnya tentang plug-in Windows, lihat Gunakan layanan EC2config untuk melakukan tugas selama peluncuran instance sistem operasi Windows lama EC2.

Jalankan Windows Sysprep dengan layanan EC2config

Gunakan prosedur berikut untuk membuat AMI standar menggunakan Windows Sysprep dan layanan EC2config.

1. Di konsol Amazon EC2, cari atau buat AMI yang ingin Anda duplikasi.

2. Luncurkan dan sambungkan ke instans Windows Anda.
3. Kustomisasikan.
4. Tentukan pengaturan konfigurasi dalam file jawaban layanan EC2Config:

   `C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml`
5. Dari menu Mulai Windows, pilih Semua Program, lalu pilih Pengaturan EC2ConfigService.
6. Pilih tab Gambar di kotak dialog Properti Layanan Ec2. Untuk informasi selengkapnya tentang opsi dan pengaturan di kotak dialog Properti Layanan Ec2, lihat Properti Layanan Ec2.
7. Pilih opsi untuk kata sandi Administrator, lalu pilih Matikan dengan Sysprep atau Matikan tanpa Sysprep. EC2Config mengubah file pengaturan berdasarkan opsi kata sandi yang Anda pilih.
   - Acak: EC2Config membuat kata sandi, mengenkripsinya dengan kunci pengguna, dan menampilkan kata sandi terenkripsi ke konsol. Kami menonaktifkan pengaturan ini setelah peluncuran pertama sehingga kata sandi ini akan tetap ada jika instans tersebut di-boot ulang atau dihentikan dan dimulai.
   - Tentukan: Kata sandi disimpan dalam file jawaban Windows Sysprep dalam bentuk tidak terenkripsi (teks yang jelas). Ketika Windows Sysprep berjalan selanjutnya, ia menetapkan kata sandi Administrator. Jika Anda mematikan sekarang, kata sandi akan segera ditetapkan. Saat layanan dimulai lagi, kata sandi Administrator dihapus. Penting untuk mengingat kata sandi ini karena Anda tidak dapat mengambilnya nanti.
   - Tetap Ada: Kata sandi yang ada untuk akun Administrator tidak berubah saat Windows Sysprep dijalankan atau EC2Config dimulai ulang. Penting untuk mengingat kata sandi ini karena Anda tidak dapat mengambilnya nanti.
8. Pilih OK.

Ketika Anda diminta untuk mengonfirmasi bahwa Anda ingin menjalankan Windows Sysprep dan mematikan instance, pilih  Ya.  Anda akan melihat bahwa EC2config menjalankan Windows Sysprep. Selanjutnya, Anda keluar dari instans, dan instansnya dimatikan. Jika Anda memeriksa halaman Instans di konsol Amazon EC2, status instan berubah dari `Running` ke `Stopping`, lalu ke `Stopped`. Ketika itu, aman untuk membuat AMI dari instans ini.

Anda dapat secara manual memanggil alat Windows Sysprep dari baris perintah menggunakan perintah berikut:

```
"%programfiles%\amazon\ec2configservice\"ec2config.exe -sysprep""
```

> **Note**
>
> Tanda kutip ganda dalam perintah tidak diperlukan jika shell CMD Anda sudah ada di direktori C:\Program Files\ Amazon\EC2ConfigService\.

Namun, Anda harus sangat berhati-hati agar opsi file XML yang ditentukan dalam folder `Ec2ConfigService\Settings` sudah benar; jika tidak, Anda mungkin tidak dapat terhubung ke instans. Untuk informasi selengkapnya tentang file pengaturan, lihat File pengaturan EC2Config. Untuk contoh mengkonfigurasi dan kemudian menjalankan Windows Sysprep dari baris perintah, lihat. `Ec2ConfigService\Scripts\InstallUpdates.ps1`

## Salin AMI Amazon EC2

Bila Anda memerlukan konfigurasi instans Amazon EC2 yang konsisten di beberapa Wilayah, Anda dapat menggunakan satu Amazon Machine Image (AMI) sebagai template untuk meluncurkan semua instans. Namun, AMI adalah sumber Region-specific daya—untuk meluncurkan instance di suatu instans tertentu Wilayah AWS, AMI harus berlokasi di Wilayah tersebut. Oleh karena itu, untuk menggunakan AMI yang sama di beberapa Wilayah, Anda harus menyalinnya dari Wilayah sumber ke setiap Wilayah target.

Metode yang Anda gunakan untuk menyalin AMI tergantung pada apakah Anda menyalin di seluruh Wilayah dalam partisi yang sama atau di partisi yang berbeda:

- Cross-Region menyalin - Salin AMI di seluruh Wilayah dalam partisi yang sama, misalnya, di seluruh Wilayah dalam partisi komersial. Metode penyalinan ini dijelaskan dalam topik ini.
- Cross-partition menyalin - Salin AMI dari satu partisi ke partisi lain, misalnya, dari partisi komersial ke AWS GovCloud (US) partisi. Untuk informasi tentang metode penyalinan ini, lihatMenyimpan dan memulihkan AMI.
- Cross-account menyalin — Buat salinan AMI yang Akun AWS telah di bagikan orang lain dengan Anda Akun AWS. Metode penyalinan ini dijelaskan dalam topik ini.

Waktu yang dibutuhkan untuk menyelesaikan operasi penyalinan untuk penyalinan AMI lintas wilayah dan lintas akun didasarkan pada upaya terbaik. Jika Anda memerlukan kontrol atas waktu penyelesaian, Anda dapat menentukan jendela penyelesaian mulai dari 15 menit hingga 48 jam, memastikan AMI Anda disalin dalam jangka waktu yang Anda butuhkan. Biaya tambahan berlaku

untuk operasi penyalinan AMI berbasis waktu. Untuk informasi selengkapnya, lihat Time-based salinan di Panduan Pengguna Amazon EBS.

Daftar Isi


- Pertimbangan-pertimbangan
- Biaya
- Berikan izin untuk menyalin AMI Amazon EC2
- Menyalin AMI
- Menghentikan operasi penyalinan AMI yang tertunda
- Cara kerja salinan AMI Amazon EC2


## Pertimbangan-pertimbangan

- Izin menyalin AMI — Anda dapat menggunakan kebijakan IAM untuk memberikan atau menolak izin pengguna untuk menyalin AMI. Mulai 28 Oktober 2024, Anda dapat menentukan izin tingkat sumber daya untuk `CopyImage` tindakan pada AMI sumber. Resource-level izin untuk AMI baru tersedia seperti sebelumnya.
- Izin peluncuran dan izin bucket Amazon S3 — AWS tidak menyalin izin peluncuran atau izin bucket Amazon S3 dari AMI sumber ke AMI baru. Setelah operasi penyalinan selesai, Anda dapat menerapkan izin peluncuran dan izin bucket Amazon S3 ke AMI yang baru.
- Tag — Anda hanya dapat menyalin tag AMI yang ditentukan pengguna yang Anda lampirkan ke AMI sumber. Tag sistem (diawali dengan `aws:`) dan tag yang ditentukan pengguna yang dilampirkan oleh Akun AWS lain tidak akan disalin. Saat menyalin AMI, Anda dapat melampirkan tag baru ke AMI baru dan snapshot pendukungnya.
- Kuota untuk salinan AMI berbasis waktu — Setelah Anda mencapai kuota throughput salinan snapshot kumulatif, permintaan salinan AMI berbasis waktu berikutnya gagal. Untuk informasi selengkapnya, lihat Kuota untuk salinan berbasis waktu di Panduan Pengguna Amazon EBS.
- Salinan tujuan sumber yang didukung — Lokasi AMI sumber menentukan apakah Anda dapat menyalinnya dan tujuan yang diizinkan untuk AMI baru:
  - Jika AMI sumber berada di Wilayah, Anda dapat menyalinnya di Wilayah tersebut, ke Wilayah lain, ke Pos Terdepan yang terkait dengan Wilayah tersebut, atau ke Zona Lokal di Wilayah tersebut.

  - Jika AMI sumber berada di Zona Lokal, Anda dapat menyalinnya di dalam Zona Lokal tersebut, ke Wilayah induk Zona Lokal tersebut, atau ke Zona Lokal tertentu lainnya dengan Wilayah induk yang sama.
  - Jika AMI sumber ada di Pos Terdepan, Anda tidak dapat menyalinnya.
- Parameter CLI untuk sumber dan tujuan — Saat menggunakan CLI, parameter berikut didukung untuk menentukan lokasi sumber AMI untuk disalin dan tujuan AMI baru. Perhatikan bahwa operasi penyalinan harus dimulai di Wilayah tujuan; jika Anda menghilangkan `--region` parameter, tujuan mengasumsikan Wilayah default yang dikonfigurasi dalam AWS CLI pengaturan Anda.

| Sumber ke tujuan | Parameter sumber | Parameter tujuan |
|---|---|---|
| Wilayah ke Wilayah | `--source-region` | `--region` |
| Wilayah ke Pos Terdepan | `--source-region` | `--destination-outpost-arn` (ARN dari Pos Terdepan) |
| Wilayah ke Zona Lokal | `--source-region`<br>Harus Daerah induk dari Zona Lokal. | `--destination-availability-zone` (nama Zona Lokal) atau `--destination-availability-zone-id` (ID Zona Lokal) |
| Zona Lokal ke Wilayah | `--source-region`<br>Harus Daerah induk dari Zona Lokal.<br>Sumber Zona Lokal diasumsikan dari lokasi ID AMI sumber yang ditentukan. | `--region`<br>Harus Daerah induk dari Zona Lokal. |
| Zona Lokal ke Zona Lokal | `--source-region`<br>Harus Daerah induk dari Zona Lokal. | `--destination-availability-zone` (nama Zona Lokal) atau `--destination-availability-zone-id` (ID Zona Lokal) |

| Sumber ke tujuan | Parameter sumber | Parameter tujuan |
| --- | --- | --- |
| | Sumber Zona Lokal diasumsikan dari lokasi ID AMI sumber yang ditentukan. | |

## Biaya

Tidak ada biaya untuk menyalin AMI jika tidak ada waktu penyelesaian yang ditentukan. Namun, biaya tambahan berlaku untuk operasi penyalinan AMI berbasis waktu. Untuk informasi selengkapnya, lihat Time-based salinan di Panduan Pengguna Amazon EBS.

Tarif penyimpanan dan transfer data standar berlaku. Jika Anda menyalin EBS-backed AMI, Anda akan dikenakan biaya untuk penyimpanan snapshot EBS tambahan.

## Berikan izin untuk menyalin AMI Amazon EC2

Untuk menyalin EBS-backed atau Amazon S3-backed AMI, Anda memerlukan izin IAM berikut:

- `ec2:CopyImage`— Untuk menyalin AMI. Untuk EBS-backed AMI, ini juga memberikan izin untuk menyalin snapshot pendukung AMI.
- `ec2:CreateTags`— Untuk menandai AMI target. Untuk EBS-backed AMI, ini juga memberikan izin untuk menandai snapshot pendukung AMI target.

Jika menyalin AMI instans yang didukung disimpan, Anda memerlukan izin IAM tambahan berikut:

- `s3:CreateBucket`— Untuk membuat bucket S3 di Wilayah target untuk AMI baru
- `s3:PutBucketOwnershipControls`- Untuk mengaktifkan ACL untuk bucket S3 yang baru dibuat sehingga objek dapat ditulis dengan ACL yang sudah di `aws-exec-read` kalengkan
- `s3:GetBucketAcl`— Untuk membaca ACL untuk bucket sumber
- `s3:ListAllMyBuckets`— Untuk menemukan bucket S3 yang ada untuk AMI di Wilayah target
- `s3:GetObject`— Untuk membaca objek di bucket sumber
- `s3:PutObject`— Untuk menulis objek di bucket target
- `s3:PutObjectAcl`- Untuk menulis izin untuk objek baru di bucket target

> **Note**
>
> Mulai 28 Oktober 2024, Anda dapat menentukan izin tingkat sumber daya untuk `CopyImage` tindakan pada AMI sumber. Resource-level izin untuk AMI target tersedia seperti sebelumnya. Untuk informasi selengkapnya, lihat tabel CopyImage di bawah Tind akan yang ditentukan oleh Amazon EC2 di Refer ensi O torisasi Layanan.

Contoh kebijakan IAM untuk menyalin EBS-backed AMI dan menandai AMI target dan snapshot

Kebijakan contoh berikut memberi Anda izin untuk menyalin EBS-backed AMI apa pun dan menandai AMI target dan snapshot pendukungnya.

> **Note**
>
> Mulai 28 Oktober 2024, Anda dapat menentukan snapshot di `Resource` elemen. Untuk informasi selengkapnya, lihat tabel CopyImage di bawah Tind akan yang ditentukan oleh Amazon EC2 di Refer ensi O torisasi Layanan.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
        "Sid": "PermissionToCopyAllImages",
        "Effect": "Allow",
        "Action": [
            "ec2:CopyImage",
            "ec2:CreateTags"
        ],
        "Resource": [
            "arn:aws:ec2:*::image/*",
            "arn:aws:ec2:*::snapshot/*"
        ]
    }]
}
```

Contoh kebijakan IAM untuk menyalin EBS-backed AMI tetapi menolak memberi tag pada snapshot baru

Iz `ec2:CopySnapshot` in diberikan secara otomatis ketika Anda mendapatkan `ec2:CopyImage` izin. Izin untuk menandai snapshot pendukung baru dapat ditolak secara eksplisit, mengesampingkan `Allow` efek untuk tindakan tersebut`ec2:CreateTags`.

Kebijakan contoh berikut memberi Anda izin untuk menyalin EBS-backed AMI apa pun, tetapi menolak Anda untuk menandai snapshot pendukung baru dari AMI target.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": [
                "ec2:CopyImage",
                "ec2:CreateTags"
            ],
            "Resource": [
                "arn:aws:ec2:*::image/*",
                "arn:aws:ec2:*::snapshot/*"
            ]
        },
        {
            "Effect": "Deny",
            "Action": "ec2:CreateTags",
            "Resource": "arn:aws:ec2:::snapshot/*"
        }
    ]
}
```

Contoh kebijakan IAM untuk menyalin S3-backed AMI Amazon dan menandai AMI target

Kebijakan contoh berikut memberi Anda izin untuk menyalin S3-backed AMI Amazon apa pun di bucket sumber yang ditentukan ke Wilayah yang ditentukan, dan menandai AMI target.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Sid": "PermissionToCopyAllImages",
            "Effect": "Allow",
            "Action": [
                "ec2:CopyImage",
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:*::image/*"
        },
        {
            "Effect": "Allow",
            "Action": "s3:ListAllMyBuckets",
            "Resource": [
                "arn:aws:s3:::*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-source-bucket/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:GetBucketAcl",
                "s3:PutObjectAcl",
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::amis-for-111122223333-in-us-east-2-hash"
            ]
        }
    ]
}
```

Untuk menemukan Nama Sumber Daya Amazon (ARN) dari bucket sumber AMI, buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/, di panel navigasi pilih AMI, dan cari nama bucket di kolom Sumber.

> **Note**
>
> Iz `s3:CreateBucket` in hanya diperlukan saat pertama kali Anda menyalin S3-backed AMI Amazon ke Wilayah individu. Setelah itu, bucket Amazon S3 yang sudah dibuat di Wilayah tersebut akan digunakan untuk menyimpan semua AMI mendatang yang Anda salin ke Wilayah tersebut.

## Menyalin AMI

Anda dapat menyalin AMI yang Anda miliki atau AMI yang dibagikan dengan Anda dari akun lain. Untuk kombinasi sumber dan tujuan yang didukung, lihatPertimbangan-pertimbangan.

Console

Untuk menyalin AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi konsol, pilih Wilayah yang berisi AMI.
3. Di panel navigasi, pilih AMI untuk menampilkan daftar AMI yang tersedia untuk Anda di Wilayah.
4. Jika Anda tidak melihat AMI yang ingin Anda salin, pilih filter yang berbeda. Anda dapat memfilter berdasarkan AMI yang dimiliki oleh saya, Gambar pribadi, gambar publik, dan gambar yang dinonaktifkan.
5. Pilih AMI untuk disalin, lalu pilih Tind akan, Sal in AMI.
6. Pada halaman Copy Amazon Machine Image (AMI), tentukan informasi berikut:
   a. Nama salinan AMI: Nama untuk AMI baru. Anda dapat menyertakan informasi sistem operasi dalam nama karena Amazon EC2 tidak memberikan informasi ini saat menampilkan detail tentang AMI.
   b. Deskripsi salinan AMI: Secara default, deskripsi mencakup informasi tentang AMI sumber sehingga Anda dapat membedakan salinan dari aslinya. Anda dapat mengubah deskripsi ini sesuai kebutuhan.

c. Wilayah Tujuan: Wilayah untuk menyalin AMI. Untuk informasi selengkapnya, lihat Cross-Region menyalin dan Cross-account menyalin.

d. Salin tag: Pilih kotak centang ini untuk menyertakan tag AMI yang ditentukan pengguna saat menyalin AMI. Tag sistem (diawali dengan `aws:`) dan tag yang ditentukan pengguna yang dilampirkan oleh Akun AWS lain tidak akan disalin.

e. Time-based copy: Anda dapat menentukan apakah operasi penyalinan selesai dalam jangka waktu tertentu atau berdasarkan upaya terbaik, sebagai berikut:

- Untuk menyelesaikan salinan dalam jangka waktu tertentu:
  - Pilih Aktifkan penyalinan berbasis waktu.
  - Untuk Durasi Pen yel esaian, masukkan jumlah menit (dalam penambahan 15 menit) yang diizinkan untuk operasi penyalinan. Durasi penyelesaian berlaku untuk semua snapshot yang terkait dengan AMI.

    Untuk informasi selengkapnya, lihat Time-based salinan di Panduan Pengguna Amazon EBS.
- Untuk melengkapi salinan berdasarkan upaya terbaik:
  - Biarkan Aktifkan salinan berbasis waktu tidak dipilih.

f. (EBS-backed Hanya AMI) Enkripsi snapshot EBS dari salinan AMI: Pilih kotak centang ini untuk mengenkripsi snapshot target, atau mengenkripsi ulang menggunakan kunci yang berbeda. Jika enkripsi secara default diaktifkan, kotak centang Enkripsi snapshot EBS salinan AMI dipilih dan tidak dapat dihapus. Untuk informasi selengkapnya, lihat Enkripsi dan penyalinan.

g. (EBS-backed Hanya AMI) Kunci KMS: Kunci KMS yang digunakan untuk mengenkripsi snapshot target.

h. Tag: Anda dapat menandai AMI baru dan snapshot baru dengan tag yang sama, atau Anda dapat menandainya dengan tag yang berbeda.

- Untuk menandai AMI baru dan snapshot baru dengan tag yang sama, pilih Tag gambar dan snapshot bersama-sama. Tag yang sama diterapkan pada AMI baru dan setiap snapshot yang dibuat.
- Untuk menandai AMI baru dan snapshot baru dengan tag yang berbeda, pilih Tag gambar dan snapshot secara terpisah. Tag yang berbeda diterapkan ke AMI baru dan snapshot yang dibuat. Namun, perhatikan bahwa semua snapshot baru yang

dibuat mendapatkan tag yang sama; Anda tidak dapat menandai setiap snapshot baru dengan tag yang berbeda.

Untuk menambahkan tag , pilih Tambahkan tag dan masukkan kunci dan nilai tag. Ulangi hal itu untuk setiap tanda.

i. Saat Anda siap untuk menyalin AMI, pilih Sal in AMI.

Status awal AMI baru adalah `Pending`. Operasi penyalinan AMI selesai saat statusnya `Available`.

AWS CLI

Untuk menyalin AMI dari satu Wilayah ke Wilayah lain

Gunakan perintah copy-image. Anda harus menentukan Wilayah sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `--source-region`. Anda dapat menentukan Wilayah tujuan menggunakan `--region` parameter (atau menghilangkan parameter ini untuk mengasumsikan Wilayah default yang dikonfigurasi dalam AWS CLI pengaturan Anda).

```
aws ec2 copy-image \
    --source-image-id ami-0abcdef1234567890 \
    --source-region us-west-2 \
    --name my-ami \
    --region us-east-1
```

Saat mengenkripsi snapshot target selama penyalinan AMI, Anda harus menentukan parameter tambahan ini: `--encrypted` dan`--kms-key-id`.

Untuk menyalin AMI dari Wilayah ke Zona Lokal

Gunakan perintah copy-image. Anda harus menentukan sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `--source-region`. Anda menentukan tujuan Zona Lokal menggunakan `--destination-availability-zone` parameter (Anda dapat menggunakan sebagai `--destination-availability-zone-id` gantinya). Perhatikan bahwa Anda hanya dapat menyalin AMI dari Wilayah ke Zona Lokal dalam Wilayah yang sama.

```
aws ec2 copy-image \
    --source-image-id ami-0abcdef1234567890 \
    --source-region cn-north-1 \
```

```
    --destination-availability-zone cn-north-1-pkx-1a \
    --name my-ami \
    --region cn-north-1
```

Untuk menyalin AMI dari Zona Lokal ke Wilayah

Gunakan perintah copy-image. Anda harus menentukan sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `--source-region`. Anda menentukan Wilayah tujuan menggunakan `--region` parameter (atau menghilangkan parameter ini untuk mengasumsikan Wilayah default yang dikonfigurasi dalam AWS CLI pengaturan Anda). Sumber Zona Lokal diasumsikan dari lokasi ID AMI sumber yang ditentukan. Perhatikan bahwa Anda hanya dapat menyalin AMI dari Zona Lokal ke Wilayah induknya.

```
aws ec2 copy-image \
    --source-image-id ami-0abcdef1234567890 \
    --source-region cn-north-1 \
    --name my-ami \
    --region cn-north-1
```

Untuk menyalin AMI dari satu Zona Lokal ke Zona Lokal lainnya

Gunakan perintah copy-image. Anda harus menentukan sumber dan tujuan. Anda menentukan Wilayah sumber Zona Lokal menggunakan `--source-region` parameter. Anda menentukan tujuan Zona Lokal menggunakan `--destination-availability-zone` parameter (Anda dapat menggunakan sebagai `--destination-availability-zone-id` gantinya). Sumber Zona Lokal diasumsikan dari lokasi ID AMI sumber yang ditentukan. Anda menentukan Region induk dari Zona Lokal tujuan menggunakan `--region` parameter (atau menghilangkan parameter ini untuk mengasumsikan Wilayah default yang dikonfigurasi dalam AWS CLI pengaturan Anda).

```
aws ec2 copy-image \
    --source-image-id ami-0abcdef1234567890 \
    --source-region cn-north-1 \
    --destination-availability-zone cn-north-1-pkx-1a \
    --name my-ami \
    --region cn-north-1
```

PowerShell

Untuk menyalin AMI dari satu Wilayah ke Wilayah lain

Gunakan Copy-EC2Image cmdlet. Anda harus menentukan Wilayah sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `-SourceRegion`. Anda dapat menentukan Wilayah tujuan menggunakan `-Region` parameter atau Set-AWSDefaultRegion cmdlet.

```
Copy-EC2Image `
    -SourceImageId ami-0abcdef1234567890 `
    -SourceRegion us-west-2 `
    -Name my-ami `
    -Region us-east-1
```

Saat mengenkripsi snapshot target selama penyalinan AMI, Anda harus menentukan parameter tambahan ini: `-Encrypted` dan`-KmsKeyId`.

Untuk menyalin AMI dari Wilayah ke Zona Lokal

Gunakan Copy-EC2Image cmdlet. Anda harus menentukan sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `-SourceRegion`. Anda menentukan tujuan Zona Lokal menggunakan `-DestinationAvailabilityZone` parameter (Anda dapat menggunakan sebagai `-DestinationAvailabilityZoneId` gantinya). Perhatikan bahwa Anda hanya dapat menyalin AMI dari Wilayah ke Zona Lokal dalam Wilayah yang sama.

```
Copy-EC2Image `
    -SourceImageId ami-0abcdef1234567890 `
    -SourceRegion cn-north-1 `
    -DestinationAvailabilityZone cn-north-1-pkx-1a `
    -Name my-ami `
    -Region cn-north-1
```

Untuk menyalin AMI dari Zona Lokal ke Wilayah

Gunakan Copy-EC2Image cmdlet. Anda harus menentukan sumber dan tujuan. Anda menentukan Wilayah sumber menggunakan parameter `-SourceRegion`. Anda menentukan Wilayah tujuan menggunakan `-Region` parameter atau Set-AWSDefaultRegion cmdlet. Sumber Zona Lokal diasumsikan dari lokasi ID AMI sumber yang ditentukan. Perhatikan bahwa Anda hanya dapat menyalin AMI dari Zona Lokal ke Wilayah induknya.

```
Copy-EC2Image `
    -SourceImageId ami-0abcdef1234567890 `
    -SourceRegion cn-north-1 `
```

```
    -Name my-ami `
    -Region cn-north-1
```

Untuk menyalin AMI dari satu Zona Lokal ke Zona Lokal lainnya

Gunakan Copy-EC2Image cmdlet. Anda harus menentukan sumber dan tujuan. Anda menentukan Wilayah sumber Zona Lokal menggunakan `-SourceRegion` parameter. Anda menentukan tujuan Zona Lokal menggunakan `-DestinationAvailabilityZone` parameter (Anda dapat menggunakan sebagai `-DestinationAvailabilityZoneId` gantinya). Sumber Zona Lokal diasumsikan dari lokasi ID AMI sumber yang ditentukan. Anda menentukan Region induk dari Zona Lokal tujuan menggunakan `-Region` parameter atau Set-AWSDefaultRegion cmdlet.

```
Copy-EC2Image `
    -SourceImageId ami-0abcdef1234567890 `
    -SourceRegion cn-north-1 `
    -DestinationAvailabilityZone cn-north-1-pkx-1a `
    -Name my-ami `
    -Region cn-north-1
```

## Menghentikan operasi penyalinan AMI yang tertunda

Anda dapat menghentikan salinan AMI yang tertunda menggunakan prosedur berikut.

Console

Untuk menghentikan operasi penyalinan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi, pilih Wilayah tujuan dari pemilih Wilayah.
3. Di panel navigasi, pilih AMI.
4. Pilih AMI untuk berhenti menyalin, lalu pilih Tindakan , Deregister AMI.
5. Saat diminta konfirmasi, pilih Batalkan pendaftaran AMI.

AWS CLI

Untuk menghentikan operasi penyalinan AMI

Gunakan perintah deregister-image.

```
aws ec2 deregister-image --image-id ami-0abcdef1234567890
```

PowerShell

Untuk menghentikan operasi penyalinan AMI menggunakan

Gunakan Unregister-EC2Image cmdlet.

```
Unregister-EC2Image -ImageId ami-0abcdef1234567890
```

## Cara kerja salinan AMI Amazon EC2

Menyalin AMI sumber menghasilkan AMI baru yang identik tetapi berbeda yang juga kami sebut sebagai AMI target. AMI target memiliki ID AMI uniknya sendiri. Anda dapat mengubah atau membatalkan pendaftaran AMI sumber tanpa memengaruhi AMI target. Begitu juga sebaliknya.

Dengan EBS-backed AMI, setiap snapshot pendukungnya disalin ke snapshot target yang identik namun berbeda. Jika Anda menyalin AMI ke Wilayah baru, snapshot tersebut merupakan salinan lengkap (non-inkremental). Jika Anda mengenkripsi snapshot dukungan yang tidak dienkripsi atau mengenkripsinya ke kunci KMS baru, snapshot tersebut merupakan salinan lengkap (non-inkremental). Operasi penyalinan AMI berikutnya akan menghasilkan salinan inkremental dari snapshot cadangan.

Daftar Isi


- Cross-Region menyalin
- Cross-account menyalin
- Time-based Operasi penyalinan AMI
- Enkripsi dan penyalinan

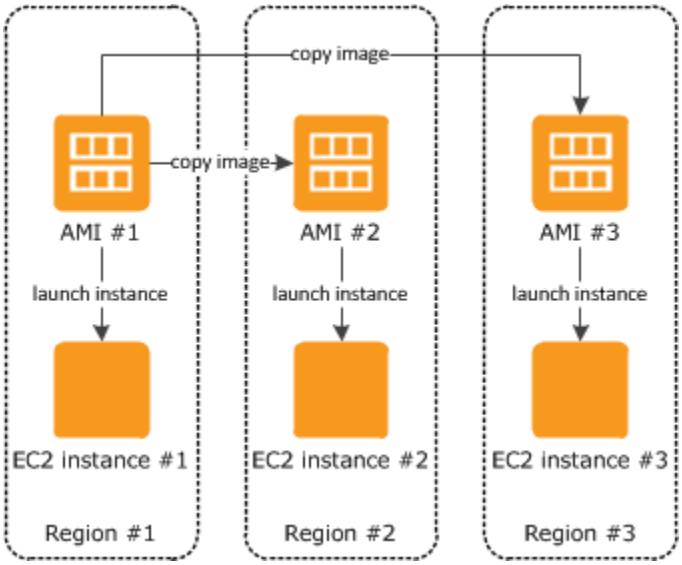

### Cross-Region menyalin

Menyalin AMI di Wilayah yang berbeda secara geografis memiliki keuntungan berikut:

- Deployment global yang konsisten: Menyalin AMI dari satu Wilayah ke Wilayah lain memungkinkan Anda meluncurkan instans yang konsisten di Wilayah yang berbeda berdasarkan AMI yang sama.
- Skalabilitas: Anda dapat dengan lebih mudah merancang dan membangun aplikasi global yang memenuhi kebutuhan pengguna Anda, terlepas dari lokasi mereka.

- Performa: Anda dapat meningkatkan performa dengan mendistribusikan aplikasi Anda, serta menemukan komponen penting pada aplikasi Anda dalam jarak yang lebih dekat dengan pengguna Anda. Anda juga dapat memanfaatkan Region-specific fitur, seperti jenis instans atau AWS layanan lainnya.
- Ketersediaan tinggi: Anda dapat merancang dan melakukan deploy aplikasi di berbagai Wilayah AWS , untuk meningkatkan ketersediaan.

Diagram berikut menunjukkan hubungan antara AMI sumber dan dua AMI yang disalin di Wilayah yang berbeda, serta instans EC2 yang diluncurkan dari masing-masing. Saat Anda meluncurkan instans dari sebuah AMI, instans tersebut berada di Wilayah yang sama dengan AMI berada. Jika Anda membuat perubahan pada AMI sumber dan ingin perubahan tersebut tercermin dalam AMI di Wilayah target, Anda harus menyalin ulang AMI sumber ke Wilayah target.




Saat pertama kali menyalin S3-backed AMI Amazon ke Wilayah, kami membuat bucket Amazon S3 untuk AMI yang disalin ke Wilayah tersebut. Semua S3-backed AMI Amazon yang Anda salin ke Wilayah tersebut disimpan dalam bucket ini. Nama bucket memiliki format berikut: amis-for- *account* -in- -*region*. *hash* Sebagai contoh: `amis-for-123456789012-in-us-east-2-yhjmxvp6`.

Prasyarat

Sebelum menyalin AMI, Anda harus memastikan konten AMI sumber telah diperbarui agar dapat berjalan di Wilayah yang berbeda. Misalnya, Anda harus memperbarui setiap string koneksi basis data atau data konfigurasi aplikasi serupa untuk mengarah ke sumber daya yang sesuai. Jika tidak, instans yang diluncurkan dari AMI baru di Wilayah tujuan mungkin masih menggunakan sumber daya dari Wilayah sumber, yang dapat memengaruhi kinerja dan biaya.

### Batasan

- Wilayah Tujuan dibatasi hingga 300 operasi penyalinan AMI bersamaan. Ini juga berlaku untuk operasi penyalinan AMI berbasis waktu.
- Anda tidak dapat menyalin AMI paravirtual (PV) ke Wilayah yang tidak mendukung AMI PV. Untuk informasi selengkapnya, lihat Tipe virtualisasi.

### Cross-account menyalin

Jika AMI dari yang lain Akun AWS di bagikan dengan Anda Akun AWS, Anda dapat menyalin AMI yang dibagikan. Ini dikenal sebagai penyalinan lintas akun. AMI yang dibagikan dengan Anda adalah sumber AMI. Saat Anda menyalin AMI sumber, Anda membuat AMI baru. AMI baru sering disebut sebagai AMI target.

### Biaya AMI

- Untuk AMI bersama, akun AMI bersama dikenakan biaya untuk penyimpanan di Wilayah.
- Jika Anda menyalin AMI yang dibagikan dengan akun Anda, Anda adalah pemilik AMI target di akun Anda.
  - Pemilik sumber AMI dikenai biaya transfer standar Amazon EBS atau Amazon S3.
  - Anda dikenakan biaya untuk penyimpanan AMI target di Wilayah tujuan.

### Izin sumber daya

Untuk menyalin AMI yang dibagikan dengan Anda dari akun lain, pemilik AMI sumber harus memberi Anda izin baca untuk penyimpanan yang mendukung AMI, bukan hanya untuk AMI itu sendiri. Penyimpanan adalah snapshot EBS terkait (untuk Amazon EBS-backed AMI) atau bucket S3 terkait (untuk S3-backed AMI Amazon). Jika AMI bersama memiliki snapshot terenkripsi, pemilik harus membagikan kunci atau kunci dengan Anda. Untuk informasi selengkapnya tentang pemberian izin sumber daya, untuk snapshot EBS, lihat Mem bagikan snapshot Amazon EBS dengan yang lain Akun AWS di Panduan Pengguna Amazon EBS, dan untuk bucket S3, lihat Manajemen Identitas dan akses untuk Amazon S3 di Panduan Pengguna Amazon S3.

> **Note**
>
> Tag yang dilampirkan ke AMI sumber tidak disalin di seluruh akun ke AMI target.

Time-based Operasi penyalinan AMI

Saat Anda memulai operasi penyalinan AMI berbasis waktu untuk EBS-backed AMI dengan satu snapshot terkait, operasi ini berperilaku dengan cara yang sama seperti operasi penyalinan snapshot berbasis waktu individual, dan batasan throughput yang sama berlaku.

Saat Anda memulai operasi penyalinan AMI berbasis waktu untuk EBS-backed AMI dengan beberapa snapshot terkait, operasi ini berperilaku dengan cara yang sama seperti operasi penyalinan snapshot berbasis waktu bersamaan, dan batasan throughput yang sama berlaku. Setiap snapshot terkait menghasilkan permintaan salinan snapshot terpisah, yang masing-masing berkontribusi pada kuota throughput salinan snapshot kumulatif Anda. Durasi penyelesaian yang Anda tentukan berlaku untuk setiap snapshot terkait.

Untuk informasi selengkapnya, lihat Time-based salinan di Panduan Pengguna Amazon EBS.

Enkripsi dan penyalinan

Tabel berikut menunjukkan dukungan enkripsi untuk berbagai AMI-copying skenario. Meskipun Anda dapat menyalin snapshot yang tidak terenkripsi untuk menghasilkan snapshot yang terenkripsi, Anda tidak dapat menyalin snapshot yang terenkripsi untuk menghasilkan snapshot yang tidak terenkripsi.

| Skenario | Deskripsi | Didukung |
|---|---|---|
| 1 | Tidak dienkripsi hingga tidak terenkripsi | Ya |
| 2 | Dienkripsi untuk dienkripsi | Ya |
| 3 | Tidak dienkripsi untuk dienkripsi | Ya |
| 4 | Dienkripsi ke tidak terenkripsi | Tidak |

> **Note**
>
> Enkripsi selama `CopyImage` tindakan hanya berlaku untuk EBS-backed AMI Amazon. Karena S3-backed AMI Amazon tidak menggunakan snapshot, Anda tidak dapat menggunakan penyalinan untuk mengubah status enkripsinya.

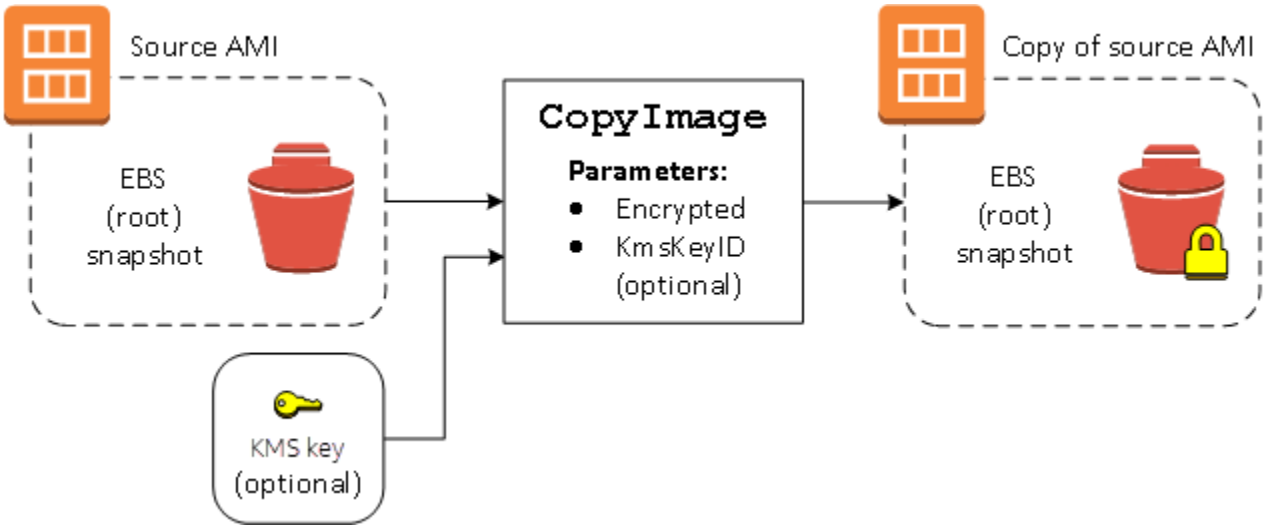
Saat Anda menyalin AMI tanpa menentukan parameter enkripsi, snapshot pendukung disalin dengan status enkripsi aslinya secara default. Oleh karena itu, jika AMI sumber didukung oleh snapshot yang tidak dienkripsi, snapshot target yang dihasilkan juga akan tidak dienkripsi. Demikian pula, jika snapshot AMI sumber dienkripsi, snapshot target yang dihasilkan juga akan dienkripsi oleh AWS KMS kunci yang sama. Untuk AMI yang didukung oleh beberapa snapshot, setiap snapshot target mempertahankan status enkripsi snapshot sumbernya yang sesuai.

Untuk mengubah status enkripsi snapshot pendukung target selama salinan AMI, Anda dapat menentukan parameter enkripsi. Contoh berikut menunjukkan kasus non-default, di mana parameter enkripsi ditentukan dengan `CopyImage` tindakan untuk mengubah status enkripsi AMI target.

Menyalin AMI sumber yang tidak terenkripsi ke AMI target yang dienkripsi

Dalam skenario ini, AMI yang didukung oleh snapshot root yang tidak dienkripsi disalin ke AMI dengan snapshot root yang dienkripsi. Tindakan `CopyImage` diinvokasi dengan dua parameter enkripsi, termasuk kunci yang dikelola konsumen. Hasilnya, status enkripsi root snapshot berubah sehingga AMI target didukung oleh snapshot root yang berisi data yang sama dengan snapshot sumber, tetapi dienkripsi menggunakan kunci yang ditentukan. Anda mengeluarkan biaya penyimpanan untuk snapshot di kedua AMI, serta biaya untuk setiap instans yang Anda luncurkan dari AMI mana pun.

> **Note**
>
> Mengaktifkan enkripsi secara default memiliki efek yang sama seperti meny `Encrypted` etel parameter `true` untuk semua snapshot di AMI.



Mengatur parameter `Encrypted` akan mengenkripsi snapshot tunggal untuk instans ini. Jika Anda tidak menentukan parameter `KmsKeyId`, kunci yang dikelola konsumen default akan digunakan untuk mengenkripsi salinan snapshot.

Untuk informasi selengkapnya tentang penyalinan AMI dengan snapshot terenkripsi, lihat Gunakan enkripsi dengan EBS-backed AMI.

## Simpan dan pulihkan AMI menggunakan S3

Anda dapat menyimpan Amazon Machine Image (AMI) dalam bucket Amazon S3, menyalin AMI ke bucket S3 lain, lalu memulihkannya dari bucket S3. Dengan menyimpan dan memulihkan AMI menggunakan bucket S3, Anda dapat menyalin AMI dari satu AWS partisi ke partisi lainnya, misalnya, dari partisi komersial utama ke AWS GovCloud (US) partisi. Anda juga dapat membuat salinan arsip AMI dengan menyimpannya dalam bucket S3.

API yang didukung untuk menyimpan dan memulihkan AMI menggunakan S3 adalah `CreateStoreImageTask`, `DescribeStoreImageTasks`, dan `CreateRestoreImageTask`.

`CopyImage`adalah API yang direkomendasikan untuk digunakan untuk menyalin AMI dalam AWS partisi. Namun, `CopyImage` tidak dapat menyalin AMI ke partisi lain.

Untuk informasi tentang AWS partisi, lihat *`partition`* halaman Nama Sumber Daya Amazon (ARN) di Panduan Pengguna IAM.

> **Warning**
>
> Pastikan Anda mematuhi semua undang-undang dan persyaratan bisnis yang berlaku saat memindahkan data antar AWS partisi atau Wil AWS ayah, termasuk, namun tidak terbatas pada, peraturan pemerintah yang berlaku dan persyaratan tempat tinggal data.

Daftar Isi


- Kasus penggunaan
- Batasan
- Biaya
- Cara kerja penyimpanan dan pemulihan AMI
- Buat tugas gambar toko

## Kasus penggunaan

Gunakan penyimpanan dan pemulihan API untuk melakukan hal berikut:

- Salin AMI antara AWS partisi
- Buat salinan arsip AMI

### Salin AMI antara AWS partisi

Dengan menyimpan dan memulihkan AMI menggunakan bucket S3, Anda dapat menyalin AMI dari satu AWS partisi ke partisi lain, atau dari satu Wil AWS ayah ke yang lain. Dalam contoh berikut, Anda menyalin AMI dari partisi komersial utama ke AWS GovCloud (US) partisi, khususnya dari Wil `us-east-2` ayah ke Wil `us-gov-east-1` ayah.

Untuk menyalin AMI dari satu partisi ke partisi lain, ikuti langkah berikut:

- Menyimpan AMI dalam bucket S3 di Wilayah saat ini dengan menggunakan `CreateStoreImageTask`. Dalam contoh ini, bucket S3 terletak di `us-east-2`.
- Pantau kemajuan tugas penyimpanan dengan menggunakan `DescribeStoreImageTasks`. Objek akan terlihat dalam bucket S3 ketika tugas selesai.
- Salin objek AMI yang tersimpan ke bucket S3 di partisi target menggunakan prosedur pilihan Anda. Dalam contoh ini, bucket S3 terletak di `us-gov-east-1`.

> **Note**
>
> Karena Anda memerlukan AWS kredentif yang berbeda untuk setiap partisi, Anda tidak dapat menyalin objek S3 langsung dari satu partisi ke partisi lainnya. Proses untuk menyalin objek S3 di seluruh partisi berada di luar lingkup dokumentasi ini. Kami menyediakan proses penyalinan berikut sebagai contoh, namun Anda harus menggunakan proses penyalinan yang memenuhi persyaratan keamanan Anda.
>
> - Untuk menyalin satu AMI antar partisi, proses penyalinan dapat sesederhana berikut ini: Unduh objek dari bucket sumber ke host perantara (misalnya, instans EC2 atau laptop), lalu unggah objek tersebut dari host perantara ke bucket target. Untuk setiap tahap proses, gunakan AWS kredenSIAL untuk partisi.
> - Untuk penggunaan yang lebih berkelanjutan, pertimbangkan untuk mengembangkan aplikasi yang mengelola salinan, yang berpotensi menggunakan unduhan dan unggahan multipart S3.

- Pulihkan AMI dari S3 bucket di partisi target dengan menggunakan `CreateRestoreImageTask`. Dalam contoh ini, bucket S3 terletak di `us-gov-east-1`.
- Pantau kemajuan tugas pemulihan dengan menggambarkan AMI untuk memeriksa kapan status menjadi tersedia. Anda juga dapat memantau persentase kemajuan dari snapshot yang membentuk AMI yang dipulihkan dengan menggambarkan snapshot.

Buat salinan arsip AMI

Anda dapat membuat salinan arsip AMI dengan menyimpannya dalam bucket S3. AMI dikemas ke dalam satu objek di S3, dan semua metadata AMI (tidak termasuk berbagi informasi) dipertahankan sebagai bagian dari AMI yang disimpan. Data AMI dikompresi sebagai bagian dari proses penyimpanan. AMI yang berisi data yang mudah dikompresi akan menghasilkan objek yang lebih kecil di S3. Untuk mengurangi biaya, Anda dapat menggunakan kelas penyimpanan S3 yang lebih murah. Untuk informasi selengkapnya, lihat Kelas Penyimpanan Amazon S3 dan Harga Amazon S3

## Batasan

- Untuk menyimpan AMI, Anda Akun AWS harus memiliki AMI dan snapshot-nya, atau AMI dan snapshot-nya harus di bagikan langsung dengan akun Anda. Anda tidak dapat menyimpan AMI jika AMI tersebut hanya dibagikan secara publik.
- Hanya EBS-backed AMI yang dapat disimpan menggunakan API ini.
- AMI Paravirtual (PV) tidak didukung.
- Ukuran AMI (sebelum kompresi) yang dapat disimpan dibatasi hingga 5.000 GB.
- Kuota permintaan gambar toko: 1.200 GB pekerjaan penyimpanan (data snapshot) sedang berlangsung.
- Kuota untuk mengembalikan permintaan gambar: 600 GB pekerjaan pemulihan (data snapshot) sedang berlangsung.
- Untuk durasi tugas penyimpanan, snapshot tidak boleh dihapus dan pengguna utama IAM yang melakukan penyimpanan harus memiliki akses ke snapshot, jika tidak maka proses penyimpanan akan gagal.
- Anda tidak dapat membuat beberapa salinan AMI dalam bucket S3 yang sama.
- AMI yang disimpan dalam bucket S3 tidak dapat dipulihkan dengan ID AMI asalnya. Anda dapat memitigasi hal ini dengan menggunakan alias AMI.

- Saat ini API penyimpanan dan pemulihan hanya didukung dengan menggunakan AWS Command Line Interface, AWS SDK, dan Amazon EC2 API. Anda tidak dapat menyimpan dan memulihkan AMI menggunakan konsol Amazon EC2.

## Biaya

Ketika Anda menyimpan dan memulihkan AMI menggunakan S3, Anda akan dikenai biaya untuk layanan yang digunakan oleh API penyimpanan dan pemulihan, dan untuk transfer data. API menggunakan S3 dan API EBS Direct (digunakan secara internal oleh API ini untuk mengakses data snapshot). Untuk informasi selengkapnya, lihat harga Amazon S3 dan harga Amazon EBS.

## Cara kerja penyimpanan dan pemulihan AMI

Untuk menyimpan dan memulihkan AMI menggunakan S3, Anda menggunakan API berikut:

- `CreateStoreImageTask` – Menyimpan AMI dalam bucket S3
- `DescribeStoreImageTasks` – Menyediakan kemajuan tugas penyimpanan AMI
- `CreateRestoreImageTask` – Memulihkan AMI dari bucket S3

Cara kerja API

- CreateStoreImageTask
- DescribeStoreImageTasks
- CreateRestoreImageTask
- Jalur file

CreateStoreImageTask

`CreateStoreImageTask`API menyimpan AMI sebagai objek tunggal dalam bucket S3.

API menciptakan tugas yang membaca semua data dari AMI dan snapshot-nya, lalu menggunakan Unggahan multipart S3 untuk menyimpan data dalam objek S3. API mengambil semua komponen AMI, termasuk sebagian besar metadata non- Region-specific AMI, dan semua snapshot EBS yang terkandung dalam AMI, dan mengemasnya ke dalam satu objek di S3. Data dikompresi sebagai bagian dari proses unggahan untuk mengurangi jumlah ruang yang digunakan di S3, sehingga objek di S3 mungkin lebih kecil dari jumlah ukuran snapshot di AMI.

Jika terlihat ada tag AMI dan snapshot ke akun yang memanggil API ini, tag tersebut dipertahankan.

Objek di S3 memiliki ID yang sama dengan AMI, tetapi dengan ekstensi `.bin`. Data berikut ini juga disimpan sebagai tag metadata S3 pada objek S3: nama AMI, deskripsi AMI, tanggal pendaftaran AMI, akun pemilik AMI, dan stempel waktu untuk operasi penyimpanan.

Waktu yang diperlukan untuk menyelesaikan tugas tergantung pada ukuran AMI. Hal ini juga bergantung pada berapa banyak tugas lain yang berlangsung karena tugas diantrekan. Anda dapat melacak kemajuan tugas dengan memanggil `DescribeStoreImageTasks` API.

Jumlah ukuran semua AMI yang sedang berlangsung dibatasi hingga 1.200 GB data snapshot EBS per akun. Penciptaan tugas lebih lanjut akan ditolak sampai tugas yang sedang berlangsung kurang dari batasan tersebut. Misalnya, jika AMI dengan 200 GB data snapshot dan AMI lain dengan 400 GB data snapshot saat ini sedang disimpan, permintaan lain akan diterima, karena total yang sedang berlangsung adalah 600 GB, yang kurang dari batas. Tetapi jika satu AMI dengan 1.200 GB data snapshot saat ini sedang disimpan, tugas lebih lanjut ditolak sampai tugas selesai.

DescribeStoreImageTasks

`DescribeStoreImageTasks`API menjelaskan kemajuan tugas penyimpanan AMI. Anda dapat menggambarkan tugas untuk AMI tertentu. Jika Anda tidak menentukan AMI, Anda akan mendapatkan daftar paginasi dari semua tugas penyimpanan gambar yang telah diproses dalam 31 hari terakhir.

Untuk setiap tugas AMI, respons menunjukkan jika tugas tersebut adalah `InProgress`, `Completed`, atau `Failed`. Untuk tugas `InProgress`, respons menunjukkan perkiraan kemajuan sebagai persentase.

Tugas tercantum dalam urutan kronologis terbalik.

Saat ini, hanya tugas dari bulan sebelumnya yang dapat dilihat.

CreateRestoreImageTask

`CreateRestoreImageTask`API memulai tugas yang mengembalikan AMI dari objek S3 yang sebelumnya dibuat dengan menggunakan `CreateStoreImageTask` permintaan.

Tugas pemulihan dapat dilakukan di Wilayah yang sama atau berbeda dari tempat tugas penyimpanan dilakukan.

Bucket S3 tempat objek AMI akan dipulihkan harus berada di Wilayah yang sama dengan tempat tugas pemulihan diminta. AMI akan dipulihkan di Wilayah ini.

AMI dipulihkan dengan metadata-nya, seperti nama, deskripsi, dan pemetaan perangkat blok yang sesuai dengan nilai-nilai AMI yang tersimpan. Nama harus unik untuk AMI di Wilayah untuk akun ini. Jika Anda tidak memberikan nama, AMI yang baru akan mendapat nama yang sama dengan AMI asal. AMI akan mendapat ID AMI baru yang dihasilkan pada saat proses pemulihan.

Waktu yang diperlukan untuk menyelesaikan tugas pemulihan AMI bergantung pada ukuran AMI. Hal ini juga bergantung pada berapa banyak tugas lain yang berlangsung karena tugas diantrekan. Anda dapat melihat kemajuan tugas dengan menggambarkan AMI (describe-images) atau snapshot EBS-nya (describe-snapshot). Jika tugas gagal, AMI dan snapshot akan dipindahkan ke status gagal.

Jumlah ukuran semua AMI yang sedang berlangsung dibatasi hingga 600 GB (berdasarkan ukuran setelah pemulihan) data snapshot EBS per akun. Penciptaan tugas lebih lanjut akan ditolak sampai tugas yang sedang berlangsung kurang dari batasan tersebut.

Jalur file

Anda dapat menggunakan jalur file saat menyimpan dan memulihkan AMI, dengan cara berikut:

- Saat menyimpan AMI di S3, jalur file dapat ditambahkan ke nama bucket. Secara internal, sistem memisahkan jalur dari nama bucket, lalu menambahkan jalur ke kunci objek yang dibuat untuk menyimpan AMI. Jalur objek lengkap ditampilkan dalam respons dari panggilan API.
- Saat memulihkan AMI, karena parameter kunci objek tersedia, jalur dapat ditambahkan ke awal nilai kunci objek.

Contoh: Nama bucket dengan jalur file ditambahkan

Saat Anda menyimpan AMI, tentukan jalur file setelah nama bucket.

```
amzn-s3-demo-bucket/path1/path2
```

Berikut ini adalah kunci objek yang dihasilkan.

```
path1/path2/ami-0abcdef1234567890.bin
```

Saat mengembalikan AMI, Anda menentukan nama bucket dan kunci objek. Sebagai contoh, lihat the section called “Buat tugas gambar toko”.

## Buat tugas gambar toko

Saat Anda menyimpan AMI dalam bucket S3, tugas gambar toko dibuat. Anda dapat menggunakan tugas gambar toko untuk memantau kemajuan dan hasil proses.

Daftar Isi


- Mengamankan AMI Anda
- Izin untuk menyimpan dan memulihkan AMI menggunakan S3
- Buat tugas gambar toko
- Membuat tugas mengembalikan gambar


### Mengamankan AMI Anda

Penting untuk memastikan bahwa bucket S3 dikonfigurasi dengan keamanan yang cukup untuk mengamankan konten AMI dan bahwa keamanan dipertahankan selama objek AMI berada di dalam bucket. Jika ini tidak dapat dilakukan, penggunaan API ini tidak disarankan. Pastikan bahwa akses publik ke bucket S3 tidak diperbolehkan. Sebaiknya aktifkan Server-side enkripsi untuk bucket S3 tempat Anda menyimpan AMI, meskipun tidak diperlukan.

Untuk informasi tentang cara mengatur pengaturan keamanan yang sesuai untuk bucket S3 Anda, tinjau topik keamanan berikut:

- Memblokir akses publik ke penyimpanan Amazon S3 Anda
- Mengatur perilaku enkripsi sisi server default untuk bucket Amazon S3
- Kebijakan bucket S3 apa yang dapat saya gunakan untuk mematuhi AWS Config aturan s3-bucket-ssl-requests-only?
- Mengaktifkan logging akses server Amazon S3

Ketika snapshot AMI disalin ke objek S3, data kemudian disalin melalui koneksi TLS. Anda dapat menyimpan AMI dengan snapshot terenkripsi, tetapi snapshot didekripsi sebagai bagian dari proses penyimpanan.

### Izin untuk menyimpan dan memulihkan AMI menggunakan S3

Jika pengguna utama IAM Anda akan menyimpan atau memulihkan AMI menggunakan Amazon S3, Anda harus memberi mereka izin yang diperlukan.

Contoh kebijakan berikut ini mencakup semua tindakan yang diperlukan untuk memungkinkan pengguna utama IAM melaksanakan tugas penyimpanan dan pemulihan.

Anda juga dapat membuat kebijakan IAM yang memberikan akses kepada pengguna utama hanya ke sumber daya tertentu. Untuk contoh kebijakan lainnya, lihat Manajemen akses untuk AWS sumber daya di Panduan Pengguna IAM.

> **Note**
>
> Jika snapshot yang membentuk AMI dienkripsi, atau jika akun Anda diaktifkan untuk enkripsi secara default, pengguna utama IAM Anda harus memiliki izin untuk menggunakan kunci KMS.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:DeleteObject",
                "s3:GetObject",
                "s3:ListBucket",
                "s3:PutObject",
                "s3:PutObjectTagging",
                "s3:AbortMultipartUpload",
                "ebs:CompleteSnapshot",
                "ebs:GetSnapshotBlock",
                "ebs:ListChangedBlocks",
                "ebs:ListSnapshotBlocks",
                "ebs:PutSnapshotBlock",
                "ebs:StartSnapshot",
                "ec2:CreateStoreImageTask",
                "ec2:DescribeStoreImageTasks",
                "ec2:CreateRestoreImageTask",
                "ec2:GetEbsEncryptionByDefault",
                "ec2:DescribeTags",
                "ec2:CreateTags"
            ],
```

```
            "Resource": "*"
        }
    ]
}
```

Buat tugas gambar toko

Untuk menyimpan AMI dalam bucket S3, mulailah dengan membuat tugas gambar toko. Waktu yang diperlukan untuk menyelesaikan tugas tergantung pada ukuran AMI. Anda dapat melacak kemajuan tugas sampai berhasil atau gagal.

AWS CLI

Untuk membuat tugas gambar toko

Gunakan perintah create-store-image-task.

```
aws ec2 create-store-image-task \
    --image-id ami-0abcdef1234567890 \
    --bucket amzn-s3-demo-bucket
```

Berikut ini adalah output contoh.

```
{
  "ObjectKey": "ami-0abcdef1234567890.bin"
}
```

Untuk menggambarkan kemajuan tugas gambar toko

Gunakan perintah describe-store-image-tasks.

```
aws ec2 describe-store-image-tasks \
    --image-ids ami-0abcdef1234567890 \
    --query StoreImageTaskResults[].StoreTaskState \
    --output text
```

Berikut ini adalah output contoh.

```
InProgress
```

PowerShell

Untuk membuat tugas gambar toko

Gunakan New-EC2StoreImageTask cmdlet.

```
New-EC2StoreImageTask `
    -ImageId ami-0abcdef1234567890 `
    -Bucket amzn-s3-demo-bucket
```

Berikut ini adalah output contoh.

```
ObjectKey         : ami-0abcdef1234567890.bin
```

Untuk menggambarkan kemajuan tugas gambar toko

Gunakan Get-EC2StoreImageTask cmdlet.

```
(Get-EC2StoreImageTask -ImageId ami-0abcdef1234567890).StoreTaskState
```

Berikut ini adalah output contoh.

```
InProgress
```

### Membuat tugas mengembalikan gambar

Anda harus menentukan nama untuk AMI yang dipulihkan. Nama harus unik untuk AMI di Wilayah untuk akun ini. AMI yang dipulihkan akan mendapat ID AMI baru.

AWS CLI

Untuk membuat tugas mengembalikan gambar

Gunakan perintah create-restore-image-task.

```
aws ec2 create-restore-image-task \
    --object-key ami-0abcdef1234567890.bin \
    --bucket amzn-s3-demo-bucket \
    --name "my-restored-ami"
```

Berikut ini adalah output contoh.

```
{
   "ImageId": "ami-1234567890abcdef0"
}
```

PowerShell

Untuk membuat tugas mengembalikan gambar

Gunakan New-EC2RestoreImageTask cmdlet.

```
New-EC2RestoreImageTask `
    -ObjectKey ami-0abcdef1234567890.bin `
    -Bucket amzn-s3-demo-bucket `
    -Name "my-restored-ami"
```

Berikut ini adalah output contoh.

```
ImageId         : ami-1234567890abcdef0
```

## Gunakan nenek moyang AMI untuk melacak asal usul AMI

Leluhur AMI membantu Anda melacak asal AMI dengan mengembalikan ID dan Wilayah dari semua AMI leluhurnya. Saat Anda membuat atau menyalin AMI, AMI baru mempertahankan ID dan Wilayah dari AMI sumbernya (induk). Ini memungkinkan Anda untuk melacak rantai AMI kembali ke root AMI.

Manfaat utama

Menggunakan leluhur AMI membantu Anda:

- Lacak derivatif AMI untuk memastikan kepatuhan terhadap kebijakan internal.
- Identifikasi AMI yang berpotensi rentan ketika masalah keamanan ditemukan di AMI leluhur.
- Pertahankan visibilitas asal AMI di beberapa Wilayah.

Topik

- Cara kerja nenek moyang AMI

- Pertimbangan-pertimbangan
- Lihat keturunan AMI
- Identifikasi sumber AMI

## Cara kerja nenek moyang AMI

Leluhur AMI mengidentifikasi AMI induk yang digunakan untuk membuat AMI yang ditentukan, induk induk, dan seterusnya, hingga AMI root. Begini cara kerjanya:

- Setiap AMI menampilkan ID dan Wilayah dari AMI sumbernya (induk).
- Dimulai dengan AMI yang Anda pilih, daftar entri leluhur menampilkan setiap AMI induk secara berurutan.
- Daftar entri leluhur menelusuri kembali hingga mencapai AMI root. Akar AMI adalah salah satu dari berikut ini:
  - AMI publik dari penyedia ter verifikasi (diidentifikasi oleh alias pemiliknya, yaitu salah satu `amazon` atau`aws-marketplace`).
  - AMI tanpa nenek moyang yang tercatat. Misalnya, saat menggunakan RegisterImage untuk membuat AMI langsung dari sekumpulan snapshot, tidak ada sumber AMI untuk dilacak, tidak seperti saat membuat AMI dari instance.
  - AMI yang sumbernya AMI berasal dari partisi yang berbeda.
  - AMI ke-50 dalam daftar. Jumlah maksimum AMI dalam daftar leluhur adalah 50.

## Pertimbangan-pertimbangan

- ID dan Wilayah dari AMI sumber hanya tersedia untuk AMI yang dibuat menggunakan CreateImage, CopyImage, atau CreateRestoreImageTask.
- Untuk AMI yang dibuat menggunakan CreateImage (membuat AMI dari instance), ID AMI sumber adalah ID AMI yang digunakan untuk meluncurkan instance.
- Informasi AMI sumber tidak tersedia untuk:
  - AMI dibuat menggunakan RegisterImage karena dibuat dari snapshot.
  - Untuk beberapa AMI yang lebih tua.
- Informasi AMI sumber dipertahankan ketika:
  - AMI disalin di seluruh Wilayah.
  - AMI sumber dideregistrasi (dihapus).

  - Anda tidak memiliki akses ke AMI sumber.
- Setiap daftar leluhur dibatasi hingga 50 AMI.

## Lihat keturunan AMI

Anda dapat melihat nenek moyang AMI menggunakan metode berikut.

### Console

Untuk melihat nenek moyang AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI dan pilih tab leluhur AMI.
4. Tabel entri leluhur AMI mencantumkan semua AMI dalam daftar leluhur.
   - ID AMI — Pengidentifikasi setiap AMI dalam daftar leluhur. Entri pertama dalam tabel adalah AMI yang dipilih, diikuti oleh leluhurnya.
   - ID AMI Sumber — ID AMI dari mana AMI di kolom ID AMI dibuat. Sebuah tanda hubung (-) menunjukkan akhir dari daftar leluhur AMI.
   - Sumber AMI Region — Wilayah AWS Tempat sumber AMI berada.
   - Tingkat leluhur — Posisi dalam daftar leluhur, di mana:
     - 0 (input AMI) menunjukkan AMI yang dipilih yang nenek moyangnya ingin Anda ketahui.
     - Jumlah yang meningkat menunjukkan nenek moyang yang lebih tua.
     - ***n***(AMI asli) menunjukkan akar AMI, dengan nomor yang menunjukkan seberapa jauh daftar leluhur berjalan.
   - Tanggal pembuatan — Saat AMI dibuat, dalam format UTC.
   - Alias pemilik — Alias pemilik AMI (misalnya,`amazon`). Tanda hubung (-) menunjukkan bahwa AMI tidak memiliki alias pemilik.

### AWS CLI

Untuk melihat nenek moyang AMI

Gunakan perintah get-image-ancestry dan tentukan ID AMI.

```
aws ec2 get-image-ancestry \
    --image-id ami-1111111111EXAMPLE \
    --region us-east-1
```

Berikut ini adalah output contoh. Output mencantumkan AMI dalam urutan leluhur: entri pertama adalah AMI (input) yang ditentukan, diikuti oleh induknya, induk induk, dan seterusnya, dan diakhiri dengan AMI root.

```
{
    "ImageAncestryEntries": [
        {
            "CreationDate": "2025-01-17T18:37:50.000Z",
            "ImageId": "ami-1111111111EXAMPLE", // Input AMI
            "SourceImageId": "ami-2222222222EXAMPLE",
            "SourceImageRegion": "us-east-1"

        },
        {
            "CreationDate": "2025-01-17T18:37:50.000Z",
            "ImageId": "ami-2222222222EXAMPLE", // Parent AMI
            "SourceImageId": "ami-3333333333EXAMPLE",
            "SourceImageRegion": "us-east-1"
        },
        ...
        {
            "CreationDate": "2025-01-17T18:37:50.000Z",
            "ImageId": "ami-8888888888EXAMPLE", // Root AMI
            "ImageOwnerAlias": "aws-marketplace",
            "SourceImageId": "ami-9999999999EXAMPLE",
            "SourceImageRegion": "us-east-2"
        }
    ]
}
```

PowerShell

Untuk melihat nenek moyang AMI

Gunakan Get-EC2ImageAncestry cmdlet.

```
Get-EC2ImageAncestry -ImageId ami-1111111111EXAMPLE
```

Berikut ini adalah output contoh. Output mencantumkan AMI dalam urutan leluhur: entri pertama adalah AMI (input) yang ditentukan, diikuti oleh induknya, induk induk, dan seterusnya, dan diakhiri dengan AMI root.

```
ImageAncestryEntries : {
    @{
        CreationDate = "2025-01-17T18:37:50.000Z"
        ImageId = "ami-1111111111EXAMPLE"    # Input AMI
        SourceImageId = "ami-2222222222EXAMPLE"
        SourceImageRegion = "us-east-1"
    },
    @{
        CreationDate = "2025-01-17T18:37:50.000Z"
        ImageId = "ami-2222222222EXAMPLE"    # Parent AMI
        SourceImageId = "ami-3333333333EXAMPLE"
        SourceImageRegion = "us-east-1"
    },
    ...
    @{
        CreationDate = "2025-01-17T18:37:50.000Z"
        ImageId = "ami-8888888888EXAMPLE"    # Root AMI
        ImageOwnerAlias = "aws-marketplace"
        SourceImageId = "ami-9999999999EXAMPLE"
        SourceImageRegion = "us-east-2"
    }
}
```

## Identifikasi sumber AMI

Jika Anda hanya perlu mengidentifikasi AMI induk langsung (sumber) yang digunakan untuk membuat AMI, Anda dapat menggunakan metode berikut.

### Console

Untuk mengidentifikasi sumber AMI yang digunakan untuk membuat AMI yang dipilih

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI untuk melihat detailnya.

   Informasi AMI sumber muncul di bidang berikut: ID AMI Sumber dan Wil ayah AMI Sumber

AWS CLI

Untuk mengidentifikasi sumber AMI yang digunakan untuk membuat AMI yang ditentukan

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --region us-east-1 \
    --image-ids ami-0abcdef1234567890 \
    --query "Images[].{ID:SourceImageId,Region:SourceImageRegion}"
```

Berikut ini adalah output contoh.

```
[
    {
        "ID": "ami-0abcdef1234567890",
        "Region": "us-west-2"
    }
]
```

PowerShell

Untuk mengidentifikasi sumber AMI yang digunakan untuk membuat AMI yang ditentukan

Gunakan Get-EC2Image cmdlet.

```
Get-EC2Image -ImageId ami-0abcdef1234567890 | Select SourceImageId,
 SourceImageRegion
```

Berikut ini adalah output contoh.

```
SourceImageId           SourceImageRegion
-------------           -----------------
ami-0abcdef1234567890 us-west-2
```

## Gunakan tanda air AMI untuk melacak dan mengidentifikasi AMI

Tanda air AMI adalah pengenal yang Anda lampirkan ke AMI pribadi Anda untuk melacak asal dan menegakkan kebijakan tata kelola. Tanda air tetap ada di seluruh siklus hidup AMI penuh:

- Jika Anda membuat AMI baru dari instance yang sedang berjalan yang diluncurkan dari AMI yang diberi tanda air, AMI baru mewarisi tanda air.
- Jika Anda menyalin AMI yang diberi tanda air, salinan tersebut membawa tanda air.
- Jika Anda membagikan AMI yang diberi tanda air dengan akun lain, tanda air tetap terlihat oleh penerima.

Manfaat utama

- Lacak asal di seluruh akun dan wilayah—Identifikasi AMI mana yang berasal dari gambar dasar yang Anda setujui.
- Filter dan temukan AMI terkait di seluruh akun Anda.
- Bantu konsumen AMI menemukan dan mengidentifikasi AMI tepercaya yang terkait dengan proyek atau organisasi.
- Menegakkan tata kelola dengan menggabungkan tanda air dengan AMI yang Diizinkan untuk membatasi peluncuran instance hanya pada AMI yang membawa tanda air yang disetujui.

Topik


- Cara kerja tanda air AMI
- Izin yang diperlukan
- Melampirkan tanda air ke AMI
- Lepaskan tanda air dari AMI
- Lihat tanda air AMI
- Filter AMI berdasarkan tanda air


## Cara kerja tanda air AMI

Tanda air AMI adalah pengidentifikasi terstruktur yang Anda lampirkan ke AMI Anda. Berikut ini menjelaskan karakteristik utama tanda air:

- Ber tahan — Saat Anda melampirkan tanda air ke AMI, tanda air tersebut diteruskan ke semua AMI turunan.
- Owner-only— Hanya pemilik AMI yang dapat melampirkan tanda air ke AMI.
- Terlihat oleh semua orang - Siapa pun yang memiliki akses ke AMI dapat melihat tanda airnya.

- Batas 5 - AMI dapat memiliki hingga total 5 tanda air.
- Tidak tersedia di AMI publik — Anda tidak dapat melampirkan tanda air ke AMI publik atau menjadikan AMI publik jika mereka memiliki tanda air.
- Disaring - Anda dapat memfilter AMI dengan tanda air saat menggunakan`describe-images`.

Format tanda air

Tanda air adalah objek terstruktur dengan bidang berikut:

- `WatermarkKey`— Pengidentifikasi unik untuk tanda air, terdiri dari*`account-id`*:*`watermark-name`*. Bagian ID akun adalah ID AWS akun 12 digit pemilik AMI. Bagian nama tanda air adalah nama yang ditentukan pelanggan.
- `SourceImageRegion`— Wilayah AMI tempat Anda awalnya melampirkan tanda air.
- `SourceImageId`— AMI yang awalnya Anda lampirkan tanda air.
- `SourceImageCreationDate`— Tanggal pembuatan AMI yang awalnya Anda lampirkan tanda air.
- `WatermarkCreationTime`— Stempel waktu saat Anda menerapkan tanda air.

Nama tanda air harus terdiri dari 3—128 karakter dan dapat berisi karakter alfanumerik, tanda kurung (()), tanda kurung persegi ([]), spasi, titik (.), garis miring (/), tanda hubung (-), tanda kutip tunggal ('), tanda at (@), atau garis bawah (_).

## Izin yang diperlukan

Untuk bekerja dengan tanda air AMI, Anda memerlukan izin IAM berikut:

- `ec2:AttachImageWatermark`— Untuk melampirkan tanda air ke AMI.
- `ec2:DetachImageWatermark`— Untuk melepaskan tanda air dari AMI.
- `ec2:DescribeImages`— Untuk melihat tanda air di AMI.

## Melampirkan tanda air ke AMI

Anda dapat melampirkan tanda air ke AMI dengan menggunakan konsol, AWS CLI, atau PowerShell.

Console

Untuk melampirkan tanda air ke AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pada tab Detail, di bagian Watermark, pilih Kelola tanda air.
5. Masukkan nama tanda air dan pilih Lampirkan.

AWS CLI

Untuk melampirkan tanda air ke AMI

Gunakan perintah attach-image-watermark.

```
aws ec2 attach-image-watermark \
    --image-id ami-1111111111EXAMPLE \
    --image-watermark-name "prod-baseline"
```

Berikut ini adalah output contoh.

```
{
    "WatermarkKey": "123456789012:prod-baseline"
}
```

PowerShell

Untuk melampirkan tanda air ke AMI

Gunakan Add-EC2ImageWatermark cmdlet.

```
Add-EC2ImageWatermark `
    -ImageId ami-1111111111EXAMPLE `
    -ImageWatermarkName "prod-baseline"
```

Anda dapat melampirkan hingga 5 tanda air ke satu AMI.

## Lepaskan tanda air dari AMI

Anda dapat melepaskan tanda air dari AMI dengan menggunakan konsol AWS CLI, atau PowerShell.

### Console

Untuk melepaskan tanda air dari AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pada tab Detail, di bagian Watermark, pilih Kelola tanda air.
5. Pilih tanda air yang akan dihapus, lalu pilih Hapus.

### AWS CLI

Untuk melepaskan tanda air dari AMI

Gunakan perintah attach -image-watermark.

```
aws ec2 detach-image-watermark \
    --image-id ami-1111111111EXAMPLE \
    --image-watermark-key "111122223333:prod-baseline"
```

### PowerShell

Untuk melepaskan tanda air dari AMI

Gunakan Remove-EC2ImageWatermark cmdlet.

```
Remove-EC2ImageWatermark `
    -ImageId ami-1111111111EXAMPLE `
    -ImageWatermarkKey "111122223333:prod-baseline"
```

> **Note**
>
> Melepaskan tanda air dari AMI tidak menghapusnya dari AMI turunan yang sudah membawa tanda air. Untuk memastikan tanda air tetap ada, berikan `ec2:DetachImageWatermark` izin hanya kepada administrator tepercaya yang perlu mengelola tanda air.

## Lihat tanda air AMI

Anda dapat melihat tanda air untuk AMI dengan menggunakan konsol, AWS CLI, atau PowerShell.

### Console

Untuk melihat tanda air untuk AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Lihat tanda air di bagian Watermark pada tab Detail.

### AWS CLI

Untuk melihat tanda air untuk AMI

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-ids ami-046863d776a820ccd \
    --region us-east-1
```

Respons mencakup `ImageWatermarks` array untuk setiap AMI.

```
{
    "Images": [
        {
            "ImageId": "ami-046863d776a820ccd",
            "Public": false,
            "OwnerId": "123456789012",
```

```
            ...
            "ImageWatermarks": [
                {
                    "WatermarkKey": "111122223333:prod-baseline",
                    "Region": "us-east-1",
                    "SourceImageId": "ami-0b752bf1df193a6c4",
                    "SourceImageCreationDate": "2024-07-10T08:15:00",
                    "CreationDate": "2024-07-12T14:30:00"
                },
                {
                    "WatermarkKey": "222222222222:security-approved",
                    "Region": "eu-north-1",
                    "SourceImageId": "ami-12345678",
                    "SourceImageCreationDate": "2024-06-01T10:00:00",
                    "CreationDate": "2024-06-05T09:45:00"
                }
            ]
        }
    ]
}
```

PowerShell

Untuk melihat tanda air untuk AMI

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image -ImageId ami-046863d776a820ccd).ImageWatermarks
```

## Filter AMI berdasarkan tanda air

Anda dapat memfilter AMI berdasarkan tanda air dengan menggunakan konsol, AWS CLI, atau PowerShell.

Console

Untuk memfilter AMI berdasarkan tanda air

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Di bilah pencarian, pilih filter tombol Watermark dan masukkan nilai kunci watermark.

AWS CLI

Untuk memfilter AMI berdasarkan tanda air

Gunakan perintah describe-images dengan filter. `image-watermark-key`

```
aws ec2 describe-images \
    --filters "Name=image-watermark-key,Values=111122223333:prod-baseline"
```

Ini mengembalikan semua AMI yang Anda akses yang membawa tanda air yang ditentukan, termasuk AMI turunan yang mewarisinya melalui operasi penyalinan.

PowerShell

Untuk memfilter AMI berdasarkan tanda air

Gunakan Get-EC2Image cmdlet dengan parameter. `-Filter`

```
Get-EC2Image `
    -Filter @{Name="image-watermark-key"; Values="111122223333:prod-baseline"}
```

Ini mengembalikan semua AMI yang Anda akses yang membawa tanda air yang ditentukan, termasuk AMI turunan yang mewarisinya melalui operasi penyalinan.

## Jenis instans yang diizinkan AMI

Sebagai pemilik AMI, Anda dapat menentukan jenis instans mana yang didukung atau tidak didukung AMI Anda. Ini mencegah peluncuran pada jenis instance yang tidak kompatibel. Amazon EC2 memberlakukan jenis instans yang diizinkan pada waktu peluncuran.

Misalnya, jika AMI Anda memerlukan perangkat keras GPU, Anda dapat membatasinya hanya untuk jenis instans GPU. Jika AMI Anda tidak kompatibel dengan keluarga instans tertentu, Anda dapat mengecualikan keluarga tersebut dari jenis instans yang diizinkan.

Daftar Isi


- Cara kerja jenis instans yang diizinkan
- Menetapkan jenis instans yang diizinkan untuk AMI
- Melihat jenis instans yang diizinkan untuk AMI
- Perilaku luncurkan dengan jenis instance yang diizinkan

- Pertimbangan-pertimbangan

## Cara kerja jenis instans yang diizinkan

Anda mengontrol jenis instans yang diizinkan melalui `InstanceTypeSpecification` atribut pada AMI. Atribut ini berisi dua daftar:

- `SupportedInstanceTypes`- Jenis instance yang didukung AMI. Hanya jenis instance ini yang dapat diluncurkan dengan AMI.
- `UnsupportedInstanceTypes`- Jenis instance yang tidak didukung AMI. Semua jenis instance lainnya dapat diluncurkan dengan AMI.

Amazon EC2 mengevaluasi spesifikasi menggunakan logika berikut:

- Jika tidak `InstanceTypeSpecification` disetel, Amazon EC2 mengizinkan semua jenis instans. Ini adalah perilaku default.
- Jika hanya `SupportedInstanceTypes` disetel, hanya jenis instance yang ditentukan yang diizinkan. Amazon EC2 memblokir semua jenis instans lainnya.
- Jika hanya `UnsupportedInstanceTypes` disetel, Amazon EC2 mengizinkan semua jenis instans kecuali yang ditentukan.
- Jika kedua daftar disetel, jenis instance harus masuk `SupportedInstanceTypes` dan tidak boleh masuk`UnsupportedInstanceTypes`.

### Dukungan wildcard

Kedua daftar mendukung pola wildcard menggunakan * karakter. Anda dapat menggunakan wildcard untuk mencocokkan beberapa jenis instans tanpa mencantumkan masing-masing satu per satu.

Tabel berikut menunjukkan contoh pola wildcard.

| Pola | Cocok |
| --- | --- |
| `t3.*` | Semua ukuran dalam keluarga `t3` instance (misalnya ,`t3.micro`,`t3.small`, dan`t3.large`) |
| `p4d.*` | Semua ukuran dalam keluarga `p4d` instance |

| Pola | Cocok |
| --- | --- |
| `g5.*` | Semua ukuran dalam keluarga g5 instance |
| `*xlarge` | Setiap jenis instans yang xlarge atau lebih besar di semua keluarga instans |
| `*.12xlarge` | Setiap jenis instance 12xlarge di semua keluarga instans |

## Menetapkan jenis instans yang diizinkan untuk AMI

Anda dapat mengatur atau mengubah jenis instans yang diizinkan untuk AMI menggunakan konsol Amazon EC2 atau AWS Command Line Interface (AWS CLI). Anda harus merupakan pemilik AMI untuk melakukan prosedur ini.

Console

Untuk menyetel jenis instans yang diizinkan untuk AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI, lalu pilih Ac tions, Kel ola spesifikasi jenis instance.
4. Di bawah Jenis instans yang didukung atau Jenis instans yang tidak didukung, pilih Tam bah.
5. Pilih jenis instans.
6. Pilih Simpan.

AWS CLI

Untuk menyetel jenis instans yang didukung untuk AMI

Gunakan perintah replace-image-instance-type-specification. Contoh berikut memungkinkan semua `t3` dan jenis `a2` instance kecualit`3.micro`.

```
aws ec2 replace-image-instance-type-specification \
    --image-id ami-1234567890abcdef0 \
    --instance-type-specification '{"SupportedInstanceTypes": ["t3.*", "a2.*"],
 "UnsupportedInstanceTypes": ["t3.micro"]}'
```

Untuk menyetel hanya jenis instans yang tidak didukung untuk AMI

Gunakan perintah berikut untuk memblokir jenis instance tertentu sambil mengizinkan semua yang lain.

```
aws ec2 replace-image-instance-type-specification \
    --image-id ami-1234567890abcdef0 \
    --instance-type-specification '{"UnsupportedInstanceTypes": ["t3.micro",
 "t3.nano"]}'
```

Untuk menghapus spesifikasi tipe instance dari AMI

Gunakan perintah berikut tanpa menentukan `--instance-type-specification` untuk menghapus batasan dan mengizinkan semua jenis instance.

```
aws ec2 replace-image-instance-type-specification \
    --image-id ami-1234567890abcdef0
```

## Melihat jenis instans yang diizinkan untuk AMI

Anda dapat melihat spesifikasi jenis instans untuk AMI menggunakan konsol Amazon EC2 atau. AWS CLI

Console

Untuk melihat jenis instans yang diizinkan untuk AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pada tab Detail, lihat jenis instans yang didukung dan tidak didukung.

AWS CLI

Untuk melihat spesifikasi tipe instans untuk AMI

Gunakan perintah describe-images. Jika AMI memiliki spesifikasi tipe instance, respons menyertakan `InstanceTypeSpecification` bidang tersebut.

```
aws ec2 describe-images \
    --image-ids ami-1234567890abcdef0
```

Berikut ini adalah contoh output untuk AMI dengan spesifikasi tipe instance.

```
{
    "Images": [
        {
            "ImageId": "ami-1234567890abcdef0",
            ...
            "InstanceTypeSpecification": {
                "SupportedInstanceTypes": [
                    {
                        "InstanceType": "t3.*"
                    },
                    {
                        "InstanceType": "a2.*"
                    }
                ],
                "UnsupportedInstanceTypes": [
                    {
                        "InstanceType": "t3.micro"
                    }
                ]
            }
        }
    ]
}
```

## Perilaku luncurkan dengan jenis instance yang diizinkan

Saat Anda meluncurkan instans, Amazon EC2 memeriksa apakah jenis instans yang ditentukan kompatibel dengan spesifikasi jenis instans AMI. Jika jenis instans tidak diizinkan, Amazon EC2 memblokir peluncuran dan mengembalikan kesalahan.

Contoh: Peluncuran yang diblokir

Perintah berikut mencoba meluncurkan `t3.micro` instance menggunakan AMI yang tidak mendukung`t3.micro`.

```
aws ec2 run-instances \
```

```
    --image-id ami-1234567890abcdef0 \
    --instance-type t3.micro
```

Amazon EC2 mengembalikan kesalahan berikut:

```
An error occurred (InvalidParameterCombination) when calling the RunInstances
 operation: This AMI does not support the specified instance type. Check DescribeImages
 for InstanceTypeSpecification, and try again.
```

Untuk mengatasi kesalahan ini, pilih jenis instance yang didukung AMI. Gunakan `describe-images` perintah untuk melihat spesifikasi tipe instance AMI.

### Pertimbangan-pertimbangan

Ingatlah informasi berikut saat Anda menggunakan jenis instans yang diizinkan.

- Secara default, AMI tidak memiliki spesifikasi tipe instance. Amazon EC2 mengizinkan semua jenis instans hingga Anda menetapkan spesifikasi secara eksplisit.
- Hanya pemilik AMI yang dapat mengatur atau mengubah spesifikasi tipe instance.
- Saat Anda menyalin AMI menggunakan`CopyImage`, Amazon EC2 mempertahankan spesifikasi jenis instans di AMI baru.
- Amazon EC2 memberlakukan spesifikasi tipe instans sebagai blok keras. Jika jenis instance tidak diizinkan, peluncuran gagal dengan `InvalidParameterCombination` kesalahan.
- Spesifikasi tidak mempengaruhi instance yang ada. Ini hanya berlaku untuk peluncuran baru.
- Peluncuran templat dan grup Penskalaan Otomatis yang mereferensikan AMI dengan spesifikasi tipe instance dapat gagal jika jenis instance yang dikonfigurasi tidak diizinkan. Sebaiknya verifikasi kompatibilitas sebelum menetapkan spesifikasi pada AMI bersama.
- T `ReplaceImageInstanceTypeSpecification` indakan menggantikan seluruh spesifikasi. Untuk menambah atau menghapus tipe instance individual, Anda harus menyertakan spesifikasi lengkap yang diperbarui dalam permintaan.

## Mengelola dan pantau penggunaan AMI

AWS menyediakan beberapa fitur untuk membantu Anda mengelola dan memantau penggunaan AMI Anda secara efektif. Anda dapat melacak akun mana yang menggunakan AMI bersama, mengidentifikasi kapan terakhir kali AMI Anda digunakan, dan menemukan sumber daya mana di akun Anda Akun AWS yang mereferensikan AMI tertentu.

Tabel berikut memberikan ikhtisar fitur untuk mengelola dan memantau penggunaan AMI:

| Fitur | Kasus penggunaan | Manfaat utama |
| --- | --- | --- |
| Laporan penggunaan AMI | Dapatkan visibilitas ke Akun AWS mana yang menggunakan AMI Anda dan berapa banyak setiap AMI digunakan. | • Identifikasi jenis sumber daya Akun AWS dan referensi AMI Anda sehingga Anda dapat membatalkan pendaftaran atau menonaktifkan AMI dengan aman.<br>• Identifikasi AMI yang tidak digunakan untuk deregistrasi untuk mengurangi biaya penyimpanan.<br>• Identifikasi AMI yang paling sering Anda gunakan. |
| Pelacakan terakhir digunakan | Periksa kapan AMI Anda terakhir digunakan. | • Identifikasi AMI yang tidak digunakan sehingga Anda dapat membatalkan pendaftaran AMI dengan aman.<br>• Identifikasi AMI yang tidak digunakan untuk deregistrasi untuk mengurangi biaya penyimpanan. |
| Pemeriksaan referensi AMI | Pastikan AWS sumber daya Anda menggunakan AMI terbaru yang sesuai. | • Mengaudit penggunaan AMI di akun Anda. |

| Fitur | Kasus penggunaan | Manfaat utama |
| --- | --- | --- |
| | | • Periksa di mana AMI tertentu direferensikan.<br>• Pertahankan kepatuhan dengan memperbarui sumber daya Anda untuk mereferensikan AMI terbaru. |

Topik


- Melihat penggunaan AMI Anda
- Periksa kapan Amazon EC2 AMI terakhir digunakan
- Mengidentifikasi sumber daya Anda yang mereferensikan AMI tertentu


## Melihat penggunaan AMI Anda

Jika Anda membagikan Gambar Mesin Amazon (AMI) dengan orang lain Akun AWS— baik dengan organisasi tertentu Akun AWS, unit organisasi (OU), atau secara publik — Anda dapat melihat bagaimana AMI ini digunakan dengan membuat laporan penggunaan AMI. Laporan memberikan visibilitas ke dalam:

- Yang Akun AWS menggunakan AMI Anda di instans EC2 atau template peluncuran
- Berapa banyak instans EC2 atau template peluncuran yang mereferensikan setiap AMI

Laporan penggunaan AMI membantu Anda mengelola AMI secara lebih efektif dengan membantu Anda:

- Identifikasi jenis sumber daya Akun AWS dan referensi AMI Anda sehingga Anda dapat membatalkan pendaftaran atau menonaktifkan AMI dengan aman.
- Identifikasi AMI yang tidak digunakan untuk deregistrasi untuk mengurangi biaya penyimpanan.
- Identifikasi AMI yang paling sering Anda gunakan.

Daftar Isi

- Cara kerja laporan penggunaan AMI
- Membuat laporan penggunaan AMI
- Lihat laporan penggunaan AMI
- Menghapus laporan penggunaan AMI
- Laporan kuota


## Cara kerja laporan penggunaan AMI

Saat Anda membuat laporan penggunaan AMI, Anda menentukan:

- AMI untuk melaporkan.
- Yang Akun AWS harus diperiksa (akun tertentu atau semua akun).
- Jenis sumber daya untuk diperiksa (instans EC2, template peluncuran, atau keduanya).
- Untuk template peluncuran, jumlah versi yang harus diperiksa (default ke 20 versi terbaru).

Amazon EC2 membuat laporan terpisah untuk setiap AMI. Setiap laporan menyediakan:

- Daftar Akun AWS penggunaan AMI.
- Hitungan sumber daya yang merujuk AMI berdasarkan jenis sumber daya per akun. Perhatikan bahwa untuk template peluncuran, jika AMI direferensikan dalam beberapa versi template peluncuran, hitungannya hanya 1.

> **Important**
>
> Saat Anda membuat laporan penggunaan AMI, laporan tersebut mungkin tidak berisi aktivitas terbaru. Aktivitas instans dari 24 jam terakhir dan aktivitas template peluncuran dari beberapa hari terakhir mungkin tidak muncul dalam laporan.

Amazon EC2 secara otomatis menghapus laporan 30 hari setelah pembuatan. Anda dapat mengunduh laporan dari konsol EC2 untuk disimpan secara lokal.

### Membuat laporan penggunaan AMI

Untuk melihat bagaimana AMI Anda digunakan, Anda harus terlebih dahulu membuat laporan penggunaan AMI, menentukan akun dan jenis sumber daya yang akan dilaporkan. Setelah laporan dibuat, Anda dapat melihat isi laporan. Anda juga dapat mengunduh laporan dari konsol EC2.

#### Console

Untuk membuat laporan penggunaan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI dan pilih T ind akan, Penggunaan AMI, Lihat penggunaan AMI saya.
4. Pada halaman Buat laporan penggunaan AMI saya, lakukan hal berikut:
   a. Untuk jenis sumber daya, pilih satu atau beberapa jenis sumber daya untuk dilaporkan.
   b. Untuk ID Akun, lakukan salah satu hal berikut:
      - Pilih Tentukan ID akun, lalu pilih Tambahkan ID akun untuk setiap akun yang akan dilaporkan.
      - Pilih Sertakan semua akun untuk melaporkan semua akun.
   c. Pilih Buat laporan penggunaan AMI saya.
5. Pada halaman AMI, pilih tab Penggunaan AMI saya.
6. Pilih ID laporan untuk melihat detailnya.

#### AWS CLI

Untuk membuat laporan penggunaan AMI untuk daftar akun

Gunakan perintah create-image-usage-report dengan parameter yang diperlukan berikut:

- `--image-id`— ID AMI untuk dilaporkan.
- `--resource-types`— Jenis sumber daya untuk diperiksa. Dalam contoh berikut, jenis sumber daya yang harus diperiksa adalah instance EC2 dan template peluncuran. Selain itu, jumlah versi template peluncuran untuk diperiksa juga ditentukan (`version-depth=100`).

Untuk melaporkan akun tertentu, gunakan `--account-ids` parameter untuk menentukan ID setiap akun yang akan dilaporkan.

```
aws ec2 create-image-usage-report \
    --image-id ami-0abcdef1234567890 \
    --account-ids 111122223333 444455556666 123456789012 \
    --resource-types ResourceType=ec2:Instance \
      'ResourceType=ec2:LaunchTemplate,ResourceTypeOptions=[{OptionName=version-
depth,OptionValues=100}]'
```

Untuk membuat laporan penggunaan AMI dari semua akun

Untuk melaporkan semua akun menggunakan AMI yang ditentukan, gunakan perintah yang sama tetapi hilangkan `--account-ids` parameter.

```
aws ec2 create-image-usage-report \
    --image-id ami-0abcdef1234567890 \
    --resource-types ResourceType=ec2:Instance \
      'ResourceType=ec2:LaunchTemplate,ResourceTypeOptions=[{OptionName=version-
depth,OptionValues=100}]'
```

Berikut ini adalah output contoh.

```
{
    "ReportId": "amiur-00b877d192f6b02d0"
}
```

Untuk memantau status pembuatan laporan

Gunakan perintah describe-image-usage-reports dan tentukan ID laporan.

```
aws ec2 describe-image-usage-reports --report-ids amiur-00b877d192f6b02d0
```

Berikut ini adalah output contoh. Nilai awal `State` bidang adalah`pending`. Untuk dapat melihat entri laporan, negara harus demikian`available`.

```
{
    "ImageUsageReports": [
        {
            "ImageId": "ami-0e9ae3dc21c2b3a64",
```

```
            "ReportId": "amiur-abcae3dc21c2b3999",
            "ResourceTypes": [
                {"ResourceType": "ec2:Instance"}
            ],
            "State": "pending",
            "CreationTime": "2025-09-29T13:27:12.322000+00:00",
            "ExpirationTime": "2025-10-28T13:27:12.322000+00:00"
        }
    ]
}
```

PowerShell

Untuk membuat laporan penggunaan AMI untuk daftar akun

Gunakan New-EC2ImageUsageReport cmdlet dengan parameter yang diperlukan berikut:

- `-ImageId`— ID AMI untuk dilaporkan.
- `-ResourceType`— Jenis sumber daya untuk diperiksa. Dalam contoh berikut, jenis sumber daya yang harus diperiksa adalah instance EC2 dan template peluncuran. Selain itu, jumlah versi template peluncuran untuk diperiksa juga ditentukan (`'version-depth' = 100`).

Untuk melaporkan akun tertentu, gunakan `-AccountId` parameter untuk menentukan ID setiap akun yang akan dilaporkan.

```
New-EC2ImageUsageReport `
    -ImageId ami-0abcdef1234567890 `
    -AccountId 111122223333 444455556666 123456789012 `
    -ResourceType @(
        @{ResourceType = 'ec2:Instance'},
        @{ResourceType = 'ec2:LaunchTemplate'ResourceTypeOptions = @{'version-depth'
 = 100}
        })
```

Untuk membuat laporan penggunaan AMI dari semua akun

Untuk melaporkan semua akun menggunakan AMI yang ditentukan, gunakan perintah yang sama tetapi hilangkan `-AccountId` parameter.

```
New-EC2ImageUsageReport `
```

```
    -ImageId ami-0abcdef1234567890 `
    -ResourceType @(
        @{ResourceType = 'ec2:Instance'},
        @{ResourceType = 'ec2:LaunchTemplate'ResourceTypeOptions = @{'version-depth'
 = 100}
        })
```

Berikut ini adalah output contoh.

```
ReportId
--------
amiur-00b877d192f6b02d0
```

Untuk memantau status pembuatan laporan

Gunakan Get-EC2ImageUsageReport perintah dan tentukan ID laporan.

```
Get-EC2ImageUsageReport -ReportId amiur-00b877d192f6b02d0
```

Berikut ini adalah output contoh. Nilai awal `State` bidang adalah`pending`. Untuk dapat melihat entri laporan, negara harus demikian`available`.

```
ImageUsageReports
-----------------
{@{ImageId=ami-0e9ae3dc21c2b3a64; ReportId=amiur-abcae3dc21c2b3999;
 ResourceTypes=System.Object[]; State=pending; CreationTime=2025-09-29;
 ExpirationTime=2025-10-28}}
```

Lihat laporan penggunaan AMI

Anda dapat melihat semua laporan penggunaan yang telah Anda buat untuk AMI dalam 30 hari terakhir. Amazon EC2 secara otomatis menghapus laporan 30 hari setelah pembuatan.

Untuk setiap laporan, Anda dapat melihat Akun AWS yang menggunakan AMI, dan untuk setiap akun, jumlah sumber daya yang merujuk AMI berdasarkan jenis sumber daya. Anda juga dapat melihat kapan pembuatan laporan dimulai. Informasi ini hanya tersedia jika laporan berada dalam status Lengkap (konsol) atau `available` (AWS CLI).

> **Important**
>
> Saat Anda membuat laporan penggunaan AMI, laporan tersebut mungkin tidak berisi aktivitas terbaru. Aktivitas instans dari 24 jam terakhir dan aktivitas template peluncuran dari beberapa hari terakhir mungkin tidak muncul dalam laporan.

### Console

Untuk melihat laporan penggunaan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pilih tab Laporan penggunaan saya.

   Daftar laporan menunjukkan:

   - Semua laporan dibuat dalam 30 hari terakhir untuk AMI yang dipilih.
   - Untuk setiap laporan, kolom Waktu yang dimulai Laporan menunjukkan tanggal laporan dibuat.
5. Pilih ID laporan untuk melihat isinya.
6. Untuk kembali ke tab Laporan penggunaan saya di halaman detail AMI, pilih Lihat semua laporan untuk AMI ini.

### AWS CLI

Untuk membuat daftar semua laporan penggunaan AMI untuk AMI yang ditentukan

Gunakan perintah describe-image-usage-reports dan tentukan ID AMI untuk mendapatkan daftar laporannya.

```
aws ec2 describe-image-usage-reports --image-ids ami-0abcdef1234567890
```

Berikut ini adalah output contoh. Setiap ID laporan terdaftar bersama dengan jenis sumber daya yang dipindai dan tanggal pembuatan dan kedaluwarsa laporan. Anda dapat menggunakan informasi ini untuk mengidentifikasi laporan yang entrinya ingin Anda lihat.

```
{
  "ImageUsageReports": [
    {
      "ImageId": "ami-0abcdef1234567890",
      "ReportId": "amiur-1111111111111111",
      "ResourceTypes": [
        {
          "ResourceType": "ec2:Instance"
        }
      ],
      "State": "available",
      "CreationTime": "2025-09-29T13:27:12.322000+00:00",
      "ExpirationTime": "2025-10-28T13:27:12.322000+00:00",
      "Tags": []
    },
    {
      "ImageId": "ami-0abcdef1234567890",
      "ReportId": "amiur-22222222222222222",
      "ResourceTypes": [
        {
          "ResourceType": "ec2:Instance"
        },
        {
          "ResourceType": "ec2:LaunchTemplate"
        }
      ],
      "State": "available",
      "CreationTime": "2025-10-01T13:27:12.322000+00:00",
      "ExpirationTime": "2025-10-30T13:27:12.322000+00:00",
      "Tags": []
    }
  ],
  "NextToken": "opaque"
}
```

Untuk melihat isi laporan penggunaan AMI untuk AMI yang ditentukan

Gunakan perintah describe-image-usage-report-entry dan tentukan ID AMI. Respons mengembalikan semua laporan untuk AMI yang ditentukan, menunjukkan akun yang telah menggunakan AMI dan jumlah sumber dayanya.

```
aws ec2 describe-image-usage-report-entries --image-ids ami-0abcdef1234567890
```

Berikut ini adalah output contoh.

```
{
  "ImageUsageReportEntries": [
    {
      "ImageId": "ami-0abcdef1234567890",
      "ResourceType": "ec2:Instance",
      "AccountId": "123412341234",
      "UsageCount": 15,
      "ReportCreationTime": "2025-09-29T13:27:12.322000+00:00",
      "ReportId": "amiur-1111111111111111"
    },
    {
      "ImageId": "ami-0abcdef1234567890",
      "ResourceType": "ec2:Instance",
      "AccountId": "123412341234",
      "UsageCount": 2,
      "ReportCreationTime": "2025-10-01T13:27:12.322000+00:00",
      "ReportId": "amiur-22222222222222222"
    },
    {
      "ImageId": "ami-0abcdef1234567890",
      "ResourceType": "ec2:Instance",
      "AccountId": "001100110011",
      "UsageCount": 39,
      "ReportCreationTime": "2025-10-01T13:27:12.322000+00:00",
      "ReportId": "amiur-22222222222222222"
    }
  ],
  "NextToken": "opaque"
}
```

Untuk melihat isi laporan penggunaan AMI untuk laporan yang ditentukan

Gunakan perintah describe-image-usage-report-entry dan tentukan ID laporan. Respons mengembalikan semua entri untuk laporan yang ditentukan, menunjukkan akun yang telah menggunakan AMI dan jumlah sumber dayanya.

```
aws ec2 describe-image-usage-report-entries --report-ids amiur-1111111111111111
```

Berikut ini adalah output contoh.

```
{
  "ImageUsageReportEntries": [
    {
      "ImageId": "ami-0abcdef1234567890",
      "ResourceType": "ec2:Instance",
      "AccountId": "123412341234",
      "UsageCount": 15,
      "ReportCreationTime": "2025-09-29T13:27:12.322000+00:00",
      "ReportId": "amiur-1111111111111111"
    },
    {
      "ImageId": "ami-0abcdef1234567890",
      "ResourceType": "ec2:LaunchTemplate",
      "AccountId": "123412341234",
      "UsageCount": 4,
      "ReportCreationTime": "2025-09-29T13:27:12.322000+00:00",
      "ReportId": "amiur-1111111111111111"
    },
    {
      "ImageId": "ami-0abcdef1234567890",
      "ResourceType": "ec2:LaunchTemplate",
      "AccountId": "001100110011",
      "UsageCount": 2,
      "ReportCreationTime": "2025-09-29T13:27:12.322000+00:00",
      "ReportId": "amiur-1111111111111111"
    }
  ],
  "NextToken": "opaque"
}
```

PowerShell

Untuk membuat daftar semua laporan penggunaan AMI untuk AMI yang ditentukan

Gunakan Get-EC2ImageUsageReport cmdlet dan tentukan ID AMI untuk mendapatkan daftar laporannya.

```
Get-EC2ImageUsageReport -ImageId ami-0abcdef1234567890
```

Berikut ini adalah output contoh. Setiap ID laporan terdaftar bersama dengan jenis sumber daya yang dipindai dan tanggal pembuatan dan kedaluwarsa laporan. Anda dapat menggunakan informasi ini untuk mengidentifikasi laporan yang entrinya ingin Anda lihat.

```
@{
    ImageUsageReports = @(
        @{
            ImageId = "ami-0abcdef1234567890"
            ReportId = "amiur-1111111111111111"
            ResourceTypes = @(
                @{
                    ResourceType = "ec2:Instance"
                }
            )
            State = "available"
            CreationTime = "2025-09-29T13:27:12.322000+00:00"
            ExpirationTime = "2025-10-28T13:27:12.322000+00:00"
        },
        @{
            ImageId = "ami-0abcdef1234567890"
            ReportId = "amiur-2222222222222222"
            ResourceTypes = @(
                @{
                    ResourceType = "ec2:Instance"
                }
            )
            State = "available"
            CreationTime = "2025-09-30T13:27:12.322000+00:00"
            ExpirationTime = "2025-10-29T13:27:12.322000+00:00"
        },
        @{
            ImageId = "ami-0abcdef1234567890"
            ReportId = "amiur-3333333333333333"
            ResourceTypes = @(
                @{
                    ResourceType = "ec2:Instance"
                }
            )
            State = "available"
            CreationTime = "2025-10-01T13:27:12.322000+00:00"
            ExpirationTime = "2025-10-30T13:27:12.322000+00:00"
        }
    )
    NextToken = "opaque"
}
```

Untuk melihat isi laporan penggunaan AMI untuk AMI yang ditentukan

Gunakan Get-EC2ImageUsageReportEntry cmdlet dan tentukan ID AMI. Respons mengembalikan semua laporan untuk AMI yang ditentukan, menunjukkan akun yang telah menggunakan AMI dan jumlah sumber dayanya.

```
Get-EC2ImageUsageReportEntry -ImageId ami-0abcdef1234567890
```

Berikut ini adalah output contoh.

```
ImageUsageReportEntries : {@{
    ImageId = "ami-0abcdef1234567890"
    ResourceType = "ec2:Instance"
    AccountId = "123412341234"
    UsageCount = 15
    ReportCreationTime = "2025-09-29T13:27:12.322000+00:00"
    ReportId = "amiur-1111111111111111"
    }, @{
    ImageId = "ami-0abcdef1234567890"
    ResourceType = "ec2:Instance"
    AccountId = "123412341234"
    UsageCount = 7
    ReportCreationTime = "2025-09-30T13:27:12.322000+00:00"
    ReportId = "amiur-22222222222222222"
    }...}
NextToken : opaque
```

Untuk melihat isi laporan penggunaan AMI untuk laporan yang ditentukan

Gunakan Get-EC2ImageUsageReportEntry cmdlet dan tentukan ID laporan. Respons mengembalikan semua entri untuk laporan yang ditentukan, menunjukkan akun yang telah menggunakan AMI dan jumlah sumber dayanya.

```
Get-EC2ImageUsageReportEntry -ReportId amiur-1111111111111111
```

Berikut ini adalah output contoh.

```
ImageUsageReportEntries : {@{
    ImageId = "ami-0abcdef1234567890"
    ResourceType = "ec2:Instance"
    AccountId = "123412341234"
    UsageCount = 15
    ReportCreationTime = "2025-09-29T13:27:12.322000+00:00"
```

```
    ReportId = "amiur-11111111111111111"
    }, @{
    ImageId = "ami-0abcdef1234567890"
    ResourceType = "ec2:LaunchTemplate"
    AccountId = "123412341234"
    UsageCount = 4
    ReportCreationTime = "2025-09-29T13:27:12.322000+00:00"
    ReportId = "amiur-11111111111111111"
    }, @{
    ImageId = "ami-0abcdef1234567890"
    ResourceType = "ec2:LaunchTemplate"
    AccountId = "************"
    UsageCount = 2
    ReportCreationTime = "2025-09-29T13:27:12.322000+00:00"
    ReportId = "amiur-11111111111111111"
    }}
NextToken : opaque
```

### Menghapus laporan penggunaan AMI

Amazon EC2 secara otomatis menghapus laporan 30 hari setelah dibuat. Anda dapat menghapusnya secara manual sebelum waktu itu.

#### Console

Untuk menghapus laporan penggunaan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pilih tab Penggunaan AMI saya.
5. Pilih tombol opsi di sebelah laporan yang akan dihapus, lalu pilih Hapus.

#### AWS CLI

Untuk menghapus laporan penggunaan AMI

Gunakan perintah delete-image-usage-report dan tentukan ID laporan.

```
aws ec2 delete-image-usage-report --report-id amiur-0123456789abcdefg
```

PowerShell

Untuk menghapus laporan penggunaan AMI

Gunakan Remove-EC2ImageUsageReport cmdlet dan tentukan ID laporan.

```
Remove-EC2ImageUsageReport -ReportId amiur-0123456789abcdefg
```

Laporan kuota

Kuota berikut berlaku untuk membuat laporan penggunaan AMI. Kuota berlaku per Wilayah AWS.

| Deskripsi | Kuota |
| --- | --- |
| In-progress (`pending`) Laporan penggunaan AMI per Akun AWS | 2.000 |
| In-progress (`pending`) Laporan penggunaan AMI per AMI | 1 |

## Periksa kapan Amazon EC2 AMI terakhir digunakan

Amazon EC2 secara otomatis melacak tanggal dan waktu ketika AMI terakhir digunakan untuk meluncurkan instance. Jika Anda memiliki AMI yang belum digunakan untuk meluncurkan instance dalam waktu yang lama, pertimbangkan apakah AMI adalah kandidat yang baik untuk deregistrasi atau penghentian. ???

Pertimbangan-pertimbangan

- Ketika AMI digunakan untuk meluncurkan instans, ada penundaan 24 jam sebelum penggunaan tersebut dilaporkan.
- Anda harus menjadi pemilik AMI untuk mendapatkan waktu peluncuran terakhir.
- Data penggunaan AMI tersedia mulai April 2017.

Console

Untuk melihat waktu peluncuran AMI terakhir

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih AMI.
3. Dari bilah filter, pilih Dimiliki oleh saya.
4. Pilih kotak centang untuk AMI.
5. Pada tab Detail, temukan Waktu peluncuran terakhir.

AWS CLI

Untuk melihat waktu peluncuran terakhir dengan menjelaskan AMI

Gunakan perintah describe-images. Jika tidak `LastLaunchedTime` ada dalam output, verifikasi bahwa Anda memiliki AMI.

```
aws ec2 describe-images \
    --image-id ami-0abcdef1234567890 \
    --query Images[].LastLaunchedTime \
    --output text
```

Berikut ini adalah output contoh.

```
2025-02-17T20:22:19Z
```

Untuk melihat atribut waktu peluncuran terakhir dari AMI

Gunakan perintah describe-image-atribut. Anda harus menjadi pemilik AMI yang ditentukan.

```
aws ec2 describe-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute lastLaunchedTime \
    --query LastLaunchedTime.Value \
    --output text
```

Berikut ini adalah output contoh.

```
2025-02-17T20:22:19Z
```

PowerShell

Untuk melihat waktu peluncuran terakhir dengan menjelaskan AMI

Gunakan Get-EC2Image cmdlet. Jika tidak `LastLaunchedTime` ada dalam output, verifikasi bahwa Anda memiliki AMI.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).LastLaunchedTime
```

Berikut ini adalah output contoh.

```
2025-02-17T20:22:19Z
```

Untuk melihat atribut waktu peluncuran terakhir dari AMI

Gunakan Get-EC2ImageAttribute cmdlet. Anda harus menjadi pemilik AMI yang ditentukan.

```
(Get-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute LastLaunchedTime).LastLaunchedTime
```

Berikut ini adalah output contoh.

```
2025-02-17T20:22:19Z
```

## Mengidentifikasi sumber daya Anda yang mereferensikan AMI tertentu

Anda dapat mengidentifikasi AWS sumber daya yang mereferensikan Gambar Mesin Amazon (AMI) tertentu, terlepas dari apakah AMI bersifat publik atau pribadi, atau siapa yang memilikinya. Visibilitas ini membantu Anda memastikan sumber daya Anda menggunakan AMI terbaru yang sesuai.

Manfaat utama

Memeriksa referensi AMI membantu Anda:

- Mengaudit penggunaan AMI di akun Anda.
- Periksa di mana AMI tertentu direferensikan.
- Pertahankan kepatuhan dengan memperbarui sumber daya Anda untuk mereferensikan AMI terbaru.

Topik


- Sumber daya yang didukung
- Cara kerja pemeriksaan referensi AMI
- Izin IAM yang diperlukan
- Langkah-langkah untuk memeriksa referensi AMI


## Sumber daya yang didukung

Referensi AMI dapat diperiksa di:

- Instans EC2
- Templat peluncuran
- Parameter SSM
- Resep gambar Pembuat Gambar
- Resep wadah Pembuat Gambar

## Cara kerja pemeriksaan referensi AMI

### Operasi dasar

Saat menjalankan pemeriksaan referensi AMI, Anda:

- Tentukan AMI mana yang akan diperiksa.
- Pilih jenis sumber daya yang akan dipindai.
- Terima daftar sumber daya Anda yang mereferensikan AMI yang ditentukan.

### Pemilihan jenis sumber daya

Di konsol, Anda memilih jenis sumber daya yang akan dipindai.

Di CLI, Anda menentukan jenis sumber daya untuk dipindai menggunakan salah satu atau kedua parameter CLI berikut:

- `IncludeAllResourceTypes`: Memindai semua jenis sumber daya yang didukung.

- `ResourceTypes`: Memindai jenis sumber daya yang Anda tentukan.

Pelingkupan respons

Anda dapat menjangkau respons untuk instans EC2 dan meluncurkan template dengan menyesuaikan `ResourceTypeOptions` nilai menggunakan parameter. `ResourceTypes` Konsol dan `IncludeAllResourceTypes` parameter keduanya menggunakan nilai opsi default. Ketika `ResourceTypes` dan `IncludeAllResourceTypes` digunakan bersama, nilai `ResourceTypes` opsi diutamakan daripada default.

Berikut ini adalah nilai default:

| Tipe sumber daya | Opsi pelingkupan () **`OptionName`** | Tujuan | Nilai default untuk **`OptionValue`** dan konsol |
|---|---|---|---|
| Instans EC2 | `state-name` | Filter berdasarkan status instance | `pending`,`running`,`shutting down` ,`terminated` ,`stopping`, `stopped` (semua negara bagian) |
| Templat peluncuran | `version-depth` | Tentukan jumlah versi template peluncuran untuk diperiksa (mulai dari versi terbaru) | `10`(versi terbaru) |

Izin IAM yang diperlukan

Untuk menggunakan DescribeImageReferences API untuk mengidentifikasi sumber daya yang mereferensikan AMI tertentu, Anda memerlukan izin IAM berikut untuk menjelaskan sumber daya:

- `ec2:DescribeInstances`
- `ec2:DescribeLaunchTemplates`
- `ec2:DescribeLaunchTemplateVersions`
- `ssm:DescribeParameters`
- `ssm:GetParameters`

- imagebuilder:ListImageRecipes
- imagebuilder:ListContainerRecipes
- imagebuilder:GetContainerRecipe

Contoh kebijakan IAM untuk menggunakan DescribeImageReferences API

Kebijakan contoh berikut memberi Anda izin untuk menggunakan DescribeImageReferences API, yang mencakup izin untuk menjelaskan instans EC2, template peluncuran, parameter Manajer Sistem, resep gambar Pembuat Gambar, dan resep wadah Pembuat Gambar.

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": "ec2:DescribeImageReferences",
   "Resource": "*"
  },
  {
   "Effect": "Allow",
   "Action": [
    "ec2:DescribeInstances",
    "ec2:DescribeLaunchTemplates",
    "ec2:DescribeLaunchTemplateVersions",
    "ssm:DescribeParameters",
    "ssm:GetParameters",
    "imagebuilder:ListImageRecipes",
    "imagebuilder:ListContainerRecipes",
    "imagebuilder:GetContainerRecipe"
   ],
   "Resource": "*",
   "Condition": {
    "ForAnyValue:StringEquals": {
     "aws:CalledVia": [
      "ec2-images.amazonaws.com"
     ]
    }
   }
  }
```

```
 ]
}
```

**Important**

Kami sangat menyarankan untuk menggunakan kebijakan AWS terkelola alih-alih membuat kebijakan sendiri. Membuat kebijakan IAM khusus yang hanya menyediakan izin yang diperlukan memerlukan waktu dan keahlian, dan akan memerlukan pembaruan saat jenis sumber daya baru tersedia. Kebijakan yang `AmazonEC2ImageReferencesAccessPolicy` dikelola:

- Memberikan semua izin yang diperlukan untuk menggunakan DescribeImageReferences API (ini termasuk izin untuk menjelaskan instans EC2, template peluncuran, parameter Manajer Sistem, dan wadah Pembuat Gambar dan resep gambar).
- Secara otomatis mendukung jenis sumber daya baru saat tersedia (terutama penting saat menggunakan `IncludeAllResourceTypes` parameter).

Anda dapat melampirkan `AmazonEC2ImageReferencesAccessPolicy` kebijakan ke identitas IAM Anda (pengguna, grup, dan peran).
Untuk melihat izin yang disertakan dalam kebijakan ini, lihat AmazonEC2ImageReferencesAccessPolicy di Referensi AWS Kebijakan Terkelola.

Langkah-langkah untuk memeriksa referensi AMI

Gunakan prosedur berikut untuk mengidentifikasi AWS sumber daya mana yang mereferensikan AMI tertentu.

Console

Untuk mengidentifikasi sumber daya yang mereferensikan AMI tertentu

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih satu atau lebih AMI untuk memeriksa referensi.
4. Pilih T indakan, Penggunaan AMI, Lihat sumber daya yang direferensikan.
5. Pada halaman Lihat sumber daya yang mereferensikan AMI yang dipilih:

a. Untuk jenis sumber daya, pilih satu atau beberapa jenis sumber daya.

b. Pilih Lihat sumber daya.

6. Bagian Sumber daya yang mereferensikan AMI yang dipilih muncul. Daftar ini menampilkan sumber daya yang mereferensikan AMI yang ditentukan. Setiap baris memberikan informasi berikut:

   - AMI ID — ID AMI yang direferensikan.
   - Jenis sumber daya — Jenis sumber daya sumber daya yang mereferensikan AMI.
   - Resource ID — ID sumber daya yang mereferensikan AMI.

AWS CLI

Untuk memeriksa referensi AMI untuk jenis sumber daya tertentu

Gunakan perintah describe-image-references dengan parameter. `--resource-types` Contoh berikut memeriksa instans EC2 (pelingkupan berdasarkan status instans), template peluncuran (cakupan hingga 20 versi template peluncuran terbaru), dan jenis sumber daya spesifik lainnya.

```
aws ec2 describe-image-references \
    --image-ids ami-0abcdef1234567890 ami-1234567890abcdef0 \
    --resource-types \
        'ResourceType=ec2:Instance,ResourceTypeOptions=[{OptionName=state-
name,OptionValues=[running,pending]}]' \
        'ResourceType=ec2:LaunchTemplate,ResourceTypeOptions=[{OptionName=version-
depth,OptionValues=[20]}]' \
        'ResourceType=ssm:Parameter' \
        'ResourceType=imagebuilder:ImageRecipe' \
        'ResourceType=imagebuilder:ContainerRecipe'
```

Berikut ini adalah output contoh.

```
{
    "ImageReferences": [
        {
            "ImageId": "ami-0abcdef1234567890",
            "ResourceType": "ec2:Instance",
            "Arn": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0"
        },
        {
```

```
            "ImageId": "ami-1234567890abcdef0",
            "ResourceType": "ec2:LaunchTemplate",
            "Arn": "arn:aws:ec2:us-east-1:123456789012:launch-template/
lt-1234567890abcdef0"
        }
    ]
}
```

Untuk memeriksa referensi AMI untuk semua jenis sumber daya yang didukung

Gunakan perintah describe-image-references dengan parameter. `--include-all-resource-types`

```
aws ec2 describe-image-references \
    --image-ids ami-0abcdef1234567890 ami-1234567890abcdef0 \
    --include-all-resource-types
```

Untuk memeriksa referensi AMI untuk semua jenis sumber daya yang didukung dan opsi tertentu

Gunakan perintah describe-image-references dengan parameter dan. `--include-all-resource-types` `--resource-types` Contoh ini memeriksa semua jenis sumber daya saat menentukan cakupan respons untuk instans EC2 terhadap instans yang sedang berjalan atau tertunda.

```
aws ec2 describe-image-references \
    --image-ids ami-0abcdef1234567890 ami-1234567890abcdef0 \
    --include-all-resource-types \
    --resource-types
 'ResourceType=ec2:Instance,ResourceTypeOptions=[{OptionName=state-
name,OptionValues=[running,pending]}]'
```

PowerShell

Untuk memeriksa referensi AMI untuk jenis sumber daya tertentu

Gunakan Get-EC2ImageReference cmdlet dengan parameter. `-ResourceType` Contoh berikut memeriksa instans EC2 (pelingkupan berdasarkan status instans), template peluncuran (cakupan hingga 20 versi template peluncuran terbaru), dan jenis sumber daya spesifik lainnya.

```
Get-EC2ImageReference `
    -ImageId 'ami-0abcdef1234567890', 'ami-1234567890abcdef0' `
```

```
    -ResourceType @(
        @{
            ResourceType = 'ec2:Instance'
            ResourceTypeOptions = @(
                @{
                    OptionName = 'state-name'
                    OptionValues = @('running', 'pending')
                }
            )
        },
        @{
            ResourceType = 'ec2:LaunchTemplate'
            ResourceTypeOptions = @(
                @{
                    OptionName = 'version-depth'
                    OptionValues = @('20')
                }
            )
        },
        @{
            ResourceType = 'ssm:Parameter'
        },
        @{
            ResourceType = 'imagebuilder:ImageRecipe'
        },
        @{
            ResourceType = 'imagebuilder:ContainerRecipe'
        }
    )
```

Untuk memeriksa referensi AMI untuk semua jenis sumber daya yang didukung

Gunakan Get-EC2ImageReference cmdlet dengan parameter. `-IncludeAllResourceTypes`

```
Get-EC2ImageReference `
    -ImageId 'ami-0abcdef1234567890', 'ami-1234567890abcdef0' `
    -IncludeAllResourceTypes
```

Untuk memeriksa referensi AMI untuk semua jenis sumber daya yang didukung dan opsi tertentu

Gunakan Get-EC2ImageReference cmdlet dengan parameter `-IncludeAllResourceTypes` dan`-ResourceType`. Contoh ini memeriksa semua jenis sumber daya saat menentukan cakupan respons untuk instans EC2 terhadap instans yang sedang berjalan atau tertunda.

```
Get-EC2ImageReference `
    -ImageId 'ami-0abcdef1234567890', 'ami-1234567890abcdef0' `
    -IncludeAllResourceTypes `
    -ResourceType @(
        @{
            ResourceType = 'ec2:Instance'
            ResourceTypeOptions = @(
                @{
                    OptionName = 'state-name'
                    OptionValues = @('running', 'pending')
                }
            )
        }
    )
```

## Menghentikan penggunaan AMI Amazon EC2

Anda dapat mengusangkan AMI untuk menunjukkan bahwa AMI sudah kedaluwarsa dan tidak boleh digunakan. Anda juga dapat menentukan tanggal pengusangan di masa depan untuk AMI, yang menunjukkan kapan AMI akan kedaluwarsa. Misalnya, Anda mungkin mengusangkan AMI yang tidak lagi dipelihara secara aktif, atau Anda mungkin mengusangkan AMI yang telah digantikan oleh versi yang lebih baru. Secara default, AMI yang sudah usang tidak muncul dalam daftar AMI, mencegah pengguna baru untuk menggunakan AMI yang telah kedaluwarsa. Namun, pengguna dan layanan peluncuran yang ada, seperti templat peluncuran dan grup Auto Scaling, dapat terus menggunakan AMI usang tersebut dengan menentukan ID-nya. Untuk menghapus AMI sehingga pengguna dan layanan tidak dapat menggunakannya, Anda harus membatalkan pendaftaran AMI.

Setelah AMI sudah usang:

- Untuk pengguna AMI, AMI yang tidak digunakan lagi tidak muncul dalam panggilan DescribeImages API kecuali Anda menentukan ID-nya atau menentukan bahwa AMI yang tidak digunakan lagi harus muncul. Pemilik AMI terus melihat AMI yang tidak digunakan lagi dalam panggilan DescribeImages API.
- Untuk pengguna AMI, AMI yang tidak digunakan lagi tidak tersedia untuk dipilih di konsol EC2. Sebagai contoh, AMI usang tidak muncul di katalog AMI dalam wizard peluncuran instans. Pemilik AMI terus melihat AMI usang di konsol EC2.

- Untuk pengguna AMI, jika Anda mengetahui ID AMI yang sudah usang, maka Anda dapat terus meluncurkan instans menggunakan AMI usang tersebut dengan menggunakan API, CLI, atau SDK.
- Layanan peluncuran, seperti templat peluncuran dan grup Auto Scaling, dapat terus mereferensikan AMI yang usang.
- Instans EC2 yang diluncurkan menggunakan AMI yang kemudian menjadi usang tidak terpengaruh, dan dapat dihentikan, dimulai, dan boot ulang.

Anda dapat membuat usang AMI privat dan publik.

Daftar Isi


- Biaya
- Pertimbangan-pertimbangan
- Membuat usang sebuah AMI
- Menggambarkan AMI yang diusangkan
- Batalkan penghentian AMI


## Biaya

Ketika Anda mengusangkan AMI, AMI tersebut tidak dihapus. Pemilik AMI terus membayar untuk snapshot AMI. Untuk berhenti membayar snapshot, pemilik AMI harus menghapus AMI dengan membatalkan pendaftaran AMI tersebut.

## Pertimbangan-pertimbangan

- Untuk mengusangkan AMI, Anda harus merupakan pemilik AMI tersebut.
- AMI yang belum digunakan baru-baru ini untuk meluncurkan instans mungkin merupakan kandidat yang baik untuk diusangkan atau dibatalkan pendaftarannya. Untuk informasi selengkapnya, lihat the section called “Periksa kapan AMI terakhir digunakan”.
- Anda dapat membuat kebijakan EBS-backed AMI Amazon Data Lifecycle Manager untuk mengotomatiskan penghentian EBS-backed AMI. Untuk informasi selengkapnya, lihat Membuat kebijakan siklus hidup AMI.
- Secara default, tanggal pengusangan semua AMI publik diatur ke dua tahun dari tanggal pembuatan AMI. Anda dapat mengatur tanggal pengusangan menjadi lebih awal dari dua tahun.

Untuk membatalkan tanggal usang, atau memundurkan tanggal usang, Anda harus menjadikan AMI privat dengan hanya membagikannya dengan akun AWS tertentu.

## Membuat usang sebuah AMI

Anda dapat mengusangkan AMI pada tanggal dan waktu tertentu. Anda harus menjadi pemilik AMI.

Batas atas untuk tanggal pengusangan adalah 10 tahun dari sekarang, kecuali untuk AMI publik, di mana batas atas adalah 2 tahun sejak tanggal pembuatan. Anda tidak dapat menentukan tanggal di masa lalu.

Console

Untuk mengusangkan AMI pada tanggal tertentu

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi kiri, pilih AMI.
3. Dari bilah filter, pilih Dimiliki oleh saya.
4. Pilih AMI, lalu pilih Tindakan, Kelola Pengusangan AMI. Anda dapat memilih beberapa AMI untuk menetapkan tanggal usang yang sama dari beberapa AMI secara sekaligus.
5. Pilih kotak centang Aktifkan, lalu masukkan tanggal dan waktu penghentian.
6. Pilih Simpan.

AWS CLI

Untuk mengusangkan AMI pada tanggal tertentu

Gunakan perintah enable-image-deprecation. Jika Anda menentukan nilai untuk detik, Amazon EC2 membulatkan detik ke menit terdekat.

```
aws ec2 enable-image-deprecation \
    --image-id ami-0abcdef1234567890 \
    --deprecate-at "2025-04-15T13:17:12.000Z"
```

PowerShell

Untuk mengusangkan AMI pada tanggal tertentu

Gunakan Enable-EC2ImageDeprecation cmdlet. Jika Anda menentukan nilai untuk detik, Amazon EC2 membulatkan detik ke menit terdekat.

```
Enable-EC2ImageDeprecation `
    -ImageId ami-0abcdef1234567890 `
    -DeprecateAt 2025-04-15T13:17:12.000Z
```

## Menggambarkan AMI yang diusangkan

Anda dapat melihat tanggal dan waktu penghentian AMI, dan memfilter AMI berdasarkan tanggal penghentian.

Console

Untuk melihat tanggal pengusangan dari AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di navigator kiri, pilih AMI, lalu pilih AMI-nya.
3. Centang bidang Waktu penghentian (jika Anda memilih kotak centang di sebelah AMI, itu terletak di tab Detail). Kolom ini menunjukkan tanggal dan waktu pengusangan AMI. Jika bidang kosong, AMI tidak usang.

Untuk memfilter AMI berdasarkan tanggal usang

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi kiri, pilih AMI.
3. Dari bilah filter, pilih Dimiliki oleh saya atau Gambar privat (gambar privat termasuk AMI yang dibagikan dengan Anda serta milik Anda).
4. Di bilah Pencarian, masukkan **`Deprecation time`** (saat Anda memasukkan huruf, akan muncul filter Waktu pengusangan), lalu pilih operator serta tanggal dan waktu.

AWS CLI

Ketika Anda menjelaskan semua AMI, hasilnya tergantung pada apakah Anda pengguna AMI atau pemilik AMI.

- Pengguna AMI — Secara default, saat Anda menjelaskan semua AMI, AMI yang tidak digunakan lagi yang dibagikan dengan Anda tetapi tidak dimiliki oleh Anda dikecualikan. Untuk menyertakan AMI yang tidak digunakan lagi dalam hasil, tentukan opsi. `--include-deprecated`
- Pemilik AMI — Saat Anda menjelaskan semua AMI, semua AMI yang Anda miliki, termasuk AMI yang tidak digunakan lagi, disertakan. Anda tidak dapat mengecualikan AMI yang tidak digunakan lagi yang Anda miliki dengan menggunakan opsi ini. `--no-include-deprecated`

Untuk menyertakan AMI yang tidak digunakan lagi saat menjelaskan semua AMI untuk akun

Gunakan perintah describe-images berikut.

```
aws ec2 describe-images
    --owners 123456789012 \
    --include-deprecated
```

Untuk menjelaskan AMI yang tidak digunakan lagi untuk akun Anda

Gunakan perintah describe-images berikut.

```
aws ec2 describe-images \
    --owners self \
    --query "Images[?DeprecationTime!=null].ImageId" \
    --output text
```

Berikut ini adalah output contoh.

```
ami-0abcdef1234567890
```

Untuk melihat tanggal pengusangan sebuah AMI

Gunakan perintah describe-images berikut. Jika tidak `DeprecationTime` ada dalam output, AMI tidak akan digunakan lagi atau disetel untuk tidak digunakan lagi pada tanggal mendatang.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
    --query Images[].DeprecationTime \
    --output text
```

Berikut ini adalah output contoh.

```
2025-05-01T00:00:00.000Z
```

PowerShell

Untuk membuat daftar AMI yang tidak digunakan lagi untuk akun Anda

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image -Owner self | Where-Object {$_.DeprecationTime -ne $null}).ImageId
```

Berikut ini adalah output contoh.

```
ami-0abcdef1234567890
```

Untuk melihat tanggal pengusangan sebuah AMI

Gunakan Get-EC2Image cmdlet. Jika tidak `DeprecationTime` ada dalam output, AMI tidak akan digunakan lagi atau disetel untuk tidak digunakan lagi pada tanggal mendatang.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).DeprecationTime
```

Berikut ini adalah output contoh.

```
2025-05-01T00:00:00.000Z
```

## Batalkan penghentian AMI

Anda dapat membatalkan penghentian AMI, yang menghapus tanggal dan waktu penghentian. Anda harus merupakan pemilik AMI untuk melakukan prosedur ini.

Console

Untuk membatalkan pengusangan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi kiri, pilih AMI.
3. Dari bilah filter, pilih Dimiliki oleh saya.
4. Pilih AMI, lalu pilih Tindakan, Kelola Pengusangan AMI. Anda dapat memilih beberapa AMI untuk membatalkan pengusangan beberapa AMI secara sekaligus.

5. Kosongkan kotak centang Aktifkan, lalu pilih S impan.

AWS CLI

Untuk membatalkan pengusangan AMI

Gunakan perintah disable- image-deprecation berikut ini.

```
aws ec2 disable-image-deprecation --image-id ami-0abcdef1234567890
```

PowerShell

Untuk membatalkan pengusangan AMI

Gunakan Disable-EC2ImageDeprecation cmdlet.

```
Disable-EC2ImageDeprecation -ImageId ami-0abcdef1234567890
```

## Nonaktifkan AMI Amazon EC2

Anda dapat menonaktifkan AMI untuk mencegahnya digunakan untuk peluncuran instans. Anda tidak dapat meluncurkan instans baru dari AMI yang dinonaktifkan. Anda dapat mengaktifkan kembali AMI yang dinonaktifkan sehingga dapat digunakan lagi untuk peluncuran instans.

Anda dapat menonaktifkan AMI privat dan publik.

Untuk mengurangi biaya penyimpanan untuk EBS-backed AMI yang dinonaktifkan yang jarang digunakan, tetapi perlu dipertahankan dalam jangka panjang, Anda dapat mengarsipkan snapshot terkait. Untuk informasi selengkapnya, lihat Mengarsipkan snapshot Amazon EBS di Panduan Pengguna Amazon EBS.

Daftar Isi


- Cara kerja penonaktifan AMI
- Biaya
- Prasyarat
- Izin IAM yang diperlukan
- Menonaktifkan AMI

- Menggambarkan AMI yang dinonaktifkan
- Re-enable AMI yang dinonaktifkan

## Cara kerja penonaktifan AMI

> **Warning**
>
> Menonaktifkan AMI akan menghapus semua izin peluncurannya.

Saat AMI dinonaktifkan:

- Status AMI berubah menjadi `disabled`.
- AMI yang dinonaktifkan tidak dapat dibagikan. Jika AMI bersifat publik atau sebelumnya dibagikan, AMI tersebut akan dijadikan privat. Jika AMI dibagikan dengan Akun AWS, organisasi, atau Unit Organisasi, mereka kehilangan akses ke AMI yang dinonaktifkan.
- AMI yang dinonaktifkan tidak muncul dalam panggilan API DescribeImages secara default.
- AMI yang dinonaktifkan tidak muncul di bawah filter konsol Dimiliki oleh saya. Untuk menemukan AMI yang dinonaktifkan, gunakan filter konsol Gambar dinonaktifkan.
- AMI yang dinonaktifkan tidak tersedia untuk dipilih untuk peluncuran instans di konsol EC2. Misalnya, AMI yang dinonaktifkan tidak muncul di katalog AMI di wizard peluncuran instans atau saat membuat templat peluncuran.
- Layanan peluncuran, seperti templat peluncuran dan grup Auto Scaling, dapat terus merujuk ke AMI yang dinonaktifkan. Peluncuran instans selanjutnya dari AMI yang dinonaktifkan akan gagal, jadi sebaiknya perbarui templat peluncuran dan grup Auto Scaling agar hanya merujuk ke AMI yang tersedia.
- Instans EC2 yang diluncurkan menggunakan AMI yang kemudian dinonaktifkan tidak terpengaruh, dan dapat dihentikan, dimulai, dan boot ulang.
- Anda tidak dapat menghapus snapshot yang terkait dengan AMI yang dinonaktifkan. Mencoba menghapus hasil snapshot terkait pada kesalahan `snapshot is currently in use`.

Saat AMI diaktifkan kembali:

- Status AMI berubah menjadi `available`, dan dapat digunakan untuk meluncurkan instans.
- AMI dapat dibagikan.

- Akun AWS, organisasi, dan Unit Organisasi yang kehilangan akses ke AMI saat dinonaktifkan tidak akan otomatis mendapatkan kembali akses, tetapi AMI dapat dibagikan lagi dengan mereka.

## Biaya

Saat Anda menonaktifkan sebuah AMI, AMI tersebut tidak dihapus. Jika AMI adalah EBS-backed AMI, Anda terus membayar untuk snapshot EBS AMI. Jika Anda ingin menyimpan AMI, Anda mungkin dapat mengurangi biaya penyimpanan dengan mengarsipkan snapshot. Untuk informasi selengkapnya, lihat Mengarsipkan snapshot Amazon EBS di Panduan Pengguna Amazon EBS. Jika Anda tidak ingin menyimpan AMI dan snapshot, Anda harus membatalkan pendaftaran AMI dan menghapus snapshot. Untuk informasi selengkapnya, lihat Membatalkan pendaftaran AMI.

## Prasyarat

Untuk menonaktifkan atau mengaktifkan kembali AMI, Anda harus menjadi pemilik AMI.

## Izin IAM yang diperlukan

Untuk menonaktifkan dan mengaktifkan kembali AMI, Anda harus memiliki izin IAM berikut:

- `ec2:DisableImage`
- `ec2:EnableImage`

## Menonaktifkan AMI

Anda dapat menonaktifkan AMI dengan menggunakan konsol EC2 atau AWS Command Line Interface (AWS CLI). Anda harus merupakan pemilik AMI untuk melakukan prosedur ini.

Console

Untuk menonaktifkan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih AMI.
3. Dari bilah filter, pilih Dimiliki oleh saya.
4. Pilih AMI, lalu pilih Tindakan, Nonaktifkan AMI. Anda dapat memilih beberapa AMI untuk dinonaktifkan sekaligus.
5. Di jendela Nonaktifkan AMI, pilih Nonaktifkan AMI.

AWS CLI

Untuk menonaktifkan AMI

Gunakan perintah berikut disable-image.

```
aws ec2 disable-image --image-id ami-0abcdef1234567890
```

PowerShell

Untuk menonaktifkan AMI

Gunakan Disable-EC2Image cmdlet.

```
Disable-EC2Image -ImageId ami-0abcdef1234567890
```

## Menggambarkan AMI yang dinonaktifkan

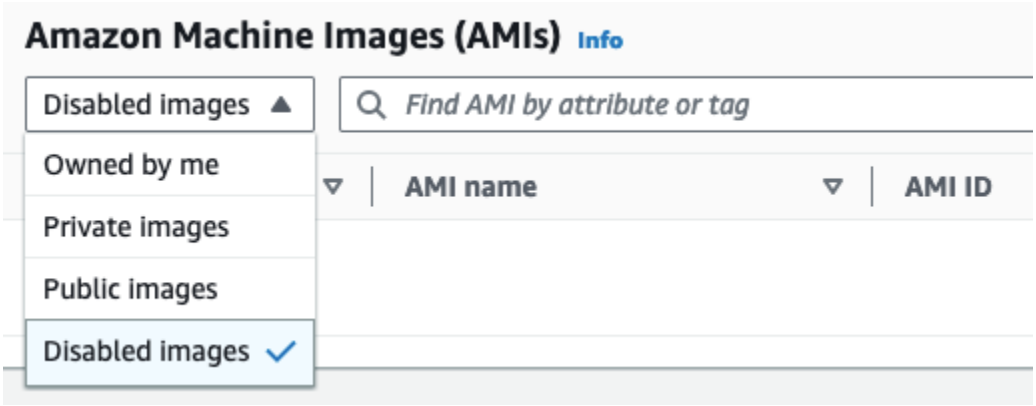
Anda dapat melihat AMI yang dinonaktifkan di konsol EC2 dan dengan menggunakan AWS CLI.

Anda harus merupakan pemilik AMI untuk melihat AMI yang dinonaktifkan. Karena AMI yang dinonaktifkan dibuat privat, Anda tidak dapat melihat AMI yang dinonaktifkan jika Anda bukan pemiliknya.

Console

Untuk melihat AMI yang dinonaktifkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih AMI.
3. Dari bilah filter, pilih Gambar  dinonaktifkan.



AWS CLI

Secara default, saat Anda menjelaskan semua AMI, AMI yang dinonaktifkan tidak disertakan dalam hasil. Untuk memasukkan AMI yang dinonaktifkan dalam hasil, tentukan `--include-disabled` opsi. Bid `State` ang untuk AMI adalah `disabled` jika AMI dinonaktifkan.

Untuk menyertakan AMI yang dinonaktifkan saat menjelaskan semua AMI untuk akun

Gunakan perintah berikut describe-images.

```
aws ec2 describe-images \
    --owners 123456789012 \
    --include-disabled
```

Untuk membuat daftar AMI yang dinonaktifkan untuk akun Anda

Gunakan perintah describe-images berikut ini.

```
aws ec2 describe-images \
    --owners self \
    --include-disabled \
    --filters Name=state,Values=disabled \
    --query Images[].ImageId \
    --output text
```

Berikut ini adalah output contoh.

```
ami-0abcdef1234567890
```

Untuk menggambarkan status AMI

Gunakan perintah berikut describe-images. Jika tidak `DeprecationTime` ada dalam output, AMI tidak akan digunakan lagi atau disetel untuk tidak digunakan lagi pada tanggal mendatang.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
    --query Images[].State \
    --output text
```

Berikut ini adalah output contoh.

```
disabled
```

PowerShell

Secara default, saat Anda menjelaskan semua AMI, AMI yang dinonaktifkan tidak disertakan dalam hasil. Untuk memasukkan AMI yang dinonaktifkan dalam hasil, tentukan `-IncludeDisabled` parameter. Bid `State` ang untuk AMI adalah `disabled` jika AMI dinonaktifkan.

Untuk membuat daftar AMI yang dinonaktifkan untuk akun Anda

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image `
    -Owner self `
    -IncludeDisabled $true | Where-Object {$_.State -eq "disabled"}).ImageId
```

Berikut ini adalah output contoh.

```
ami-0abcdef1234567890
```

Untuk menggambarkan status AMI

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).State.Value
```

Berikut ini adalah output contoh.

```
disabled
```

## Re-enable AMI yang dinonaktifkan

Anda dapat megaktifkan kembali AMI yang dinonaktifkan. Anda harus merupakan pemilik AMI untuk melakukan prosedur ini.

Console

Untuk mengaktifkan kembali AMI yang dinonaktifkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih AMI.
3. Dari bilah filter, pilih Gambar  dinonaktifkan.
4. Pilih AMI, lalu pilih Tindakan, Nonaktifkan AMI. Anda dapat memilih beberapa AMI untuk mengaktifkan kembali beberapa AMI sekaligus.
5. Di jendela Aktifkan AMI, pilih Aktifkan.

AWS CLI

Untuk mengaktifkan kembali AMI yang dinonaktifkan

Gunakan perintah berikut enable-image.

```
aws ec2 enable-image --image-id ami-0abcdef1234567890
```

PowerShell

Untuk mengaktifkan kembali AMI yang dinonaktifkan

Gunakan Enable-EC2Image cmdlet.

```
Enable-EC2Image -ImageId ami-0abcdef1234567890
```

## Batalkan pendaftaran ke Amazon EC2 AMI

Saat Anda membatalkan pendaftaran AMI, Amazon EC2 menghapusnya secara permanen. Setelah Anda membatalkan pendaftaran AMI, maka Anda tidak dapat menggunakan AMI tersebut untuk meluncurkan instans baru. Anda dapat mempertimbangkan untuk membatalkan pendaftaran AMI setelah Anda selesai menggunakannya.

Untuk melindungi dari deregistrasi AMI yang tidak disengaja atau berbahaya, Anda dapat mengaktifkan perlindungan deregistrasi. Jika Anda tidak sengaja membatalkan pendaftaran EBS-

backed AMI, Anda dapat menggunakan Recycle Bin untuk mengembalikannya hanya jika Anda mengembalikannya dalam jangka waktu yang diizinkan sebelum dihapus secara permanen.

Saat membatalkan pendaftaran AMI, Anda dapat secara opsional menghapus snapshot terkait secara bersamaan. Namun, jika snapshot dikaitkan dengan beberapa AMI, snapshot tidak akan dihapus meskipun ditentukan untuk dihapus, meskipun AMI masih akan dideregistrasi. Setiap snapshot yang tidak dihapus akan terus dikenakan biaya penyimpanan.

Menghapus pendaftaran AMI tidak berpengaruh pada instans apa pun yang diluncurkan dari AMI. Anda dapat terus menggunakan instance ini. Secara default, deregistrasi AMI juga tidak berpengaruh pada snapshot apa pun yang dibuat selama proses pembuatan AMI. Anda akan terus mengeluarkan biaya penggunaan untuk instans ini dan biaya penyimpanan untuk snapshot. Oleh karena itu, untuk menghindari biaya yang tidak perlu, kami sarankan Anda menghentikan instans apa pun dan menghapus snapshot apa pun yang tidak Anda perlukan. Anda dapat menghapus snapshot baik secara otomatis selama deregistrasi atau secara manual setelah deregistrasi. Untuk informasi selengkapnya, lihat Hindari biaya dari sumber daya yang tidak terpakai.

Untuk instans yang diluncurkan dari AMI yang kemudian dideregistrasi, Anda masih dapat melihat beberapa informasi tingkat tinggi tentang AMI dengan menggunakan perintah. `describe-instance-image-metadata` AWS CLI Untuk informasi selengkapnya, lihat describe- instance-image-metadata.

Daftar Isi


- Pertimbangan-pertimbangan
- Membatalkan pendaftaran AMI
- Hindari biaya dari sumber daya yang tidak terpakai
- Lindungi AMI Amazon EC2 dari deregistrasi


## Pertimbangan-pertimbangan

- Anda tidak dapat membatalkan pendaftaran AMI yang tidak dimiliki oleh akun Anda.
- Anda tidak dapat menggunakan Amazon EC2 untuk membatalkan pendaftaran AMI yang dikelola oleh layanan. AWS Backup Sebagai gantinya, gunakan AWS Backup untuk menghapus titik pemulihan yang sesuai di brankas cadangan. Untuk informasi selengkapnya, lihat Menghapus cadangan di dalam Panduan Developer AWS Backup .

## Membatalkan pendaftaran AMI

Anda dapat membatalkan pendaftaran EBS-backed AMI dan AMI Amazon S3-backed . Untuk EBS-backed AMI, Anda dapat secara opsional menghapus snapshot terkait secara bersamaan. Namun, jika snapshot dikaitkan dengan AMI lain, itu tidak akan dihapus meskipun ditentukan untuk dihapus.

Console

Untuk membatalkan pendaftaran AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Dari bilah filter, pilih Dim iliki oleh saya untuk mencantumkan AMI yang tersedia, atau pilih Gambar dinonaktifkan untuk mencantumkan AMI yang dinonaktifkan.
4. Pilih AMI yang akan dibatalkan pendaftaran.
5. Pilih Tindakan, Batalkan pendaftaran AMI.
6. (Opsional) Untuk menghapus snapshot terkait selama deregistrasi, pilih kotak centang H apus snapshot terkait.

> **Note**
>
> Jika snapshot dikaitkan dengan AMI lain, itu tidak dihapus, bahkan jika kotak centang dipilih.

7. Pilih Batalkan Pendaftaran AMI.

   Mungkin perlu waktu beberapa menit sebelum konsol menghapus AMI dari daftar. Pilih Segarkan untuk menyegarkan status.

AWS CLI

Untuk membatalkan pendaftaran AMI

Gunakan perintah deregister -image berikut.

```
aws ec2 deregister-image --image-id ami-0abcdef1234567890
```

Untuk membatalkan pendaftaran AMI dan menghapus snapshot terkait

Gunakan perintah deregister-image berikut dan tentukan parameternya. `--delete-associated-snapshots` Perhatikan bahwa jika snapshot dikaitkan dengan AMI lain, itu tidak dihapus, bahkan jika Anda menentukan parameter ini.

```
aws ec2 deregister-image \
    --image-id ami-0abcdef1234567890 \
    --delete-associated-snapshots
```

PowerShell

Untuk membatalkan pendaftaran AMI

Gunakan Unregister-EC2Image cmdlet.

```
Unregister-EC2Image -ImageId ami-0abcdef1234567890
```

Untuk membatalkan pendaftaran AMI dan menghapus snapshot terkait

Gunakan Unregister-EC2Image cmdlet dan tentukan parameternya. `-DeleteAssociatedSnapshots` Perhatikan bahwa jika snapshot dikaitkan dengan AMI lain, itu tidak dihapus, bahkan jika Anda menentukan parameter ini.

```
Unregister-EC2Image `
    -ImageId ami-0abcdef1234567890 `
    -DeleteAssociatedSnapshots
```

## Hindari biaya dari sumber daya yang tidak terpakai

Menghapus pendaftaran AMI tidak, secara default, menghapus semua sumber daya yang terkait dengan AMI. Sumber daya ini mencakup snapshot untuk EBS-backed AMI dan file di Amazon S3 untuk Amazon S3-backed AMI. Saat Anda membatalkan pendaftaran AMI, Anda juga tidak menghentikan atau menghentikan instans apa pun yang diluncurkan dari AMI.

Anda akan terus mengeluarkan biaya untuk menyimpan snapshot dan file, dan Anda akan dikenakan biaya untuk setiap instans yang sedang berjalan.

Untuk menghindari timbulnya jenis biaya yang tidak perlu ini, sebaiknya hapus sumber daya apa pun yang tidak Anda perlukan.

EBS-backed AMI

- Hapus snapshot terkait saat membatalkan pendaftaran AMI. Untuk informasi selengkapnya, lihat Membatalkan pendaftaran AMI.
- Jika Anda membatalkan pendaftaran AMI tanpa menghapus snapshot terkait, Anda dapat menghapus snapshot secara manual. Snapshot dari volume root instance yang dibuat selama pembuatan AMI memiliki format deskripsi berikut:

  ```
  Created by CreateImage(i-1234567890abcdef0) for ami-0abcdef1234567890
  ```

- Jika Anda tidak lagi memerlukan instans yang diluncurkan dari AMI, Anda dapat menghentikan atau menghenti kannya. Untuk mencantumkan instance, filter berdasarkan ID AMI.

S3-backed AMI Amazon

- Hapus bundel di Amazon S3 dengan menggunakan perintah ec2-delete-bundle (AMI tools).
- Jika bucket Amazon S3 kosong setelah Anda menghapus bundel, dan Anda tidak memiliki kegunaan lebih lanjut untuk bucket itu, Anda dapat menghapus bucket tersebut.
- Jika Anda tidak lagi memerlukan instans yang diluncurkan dari AMI, Anda dapat mengh enti kannya. Untuk mencantumkan instance, filter berdasarkan ID AMI.

## Lindungi AMI Amazon EC2 dari deregistrasi

Anda dapat mengaktifkan perlindungan deregistrasi pada AMI untuk mencegah pengh apusan yang tidak disengaja atau berbahaya. Saat Anda mengaktifkan perlindungan deregistrasi, AMI tidak dapat dibatalkan pendaftaran oleh pengguna mana pun, terlepas dari izin IAM mereka. Jika Anda ingin membatalkan pendaftaran AMI, Anda harus terlebih dahulu mematikan perlindungan deregistrasi di atasnya.

Saat mengaktifkan perlindungan deregistrasi pada AMI, Anda memiliki opsi untuk menyertakan periode cooldown 24 jam. Periode cooldown ini adalah waktu di mana perlindungan deregistrasi tetap berlaku setelah Anda mematikannya. Selama periode cooldown ini, AMI tidak dapat dibatalkan pendaftaran. Ketika periode cooldown berakhir, AMI dapat dideregistrasi.

Perlindungan deregistrasi dimatikan secara default pada semua AMI yang ada dan baru.

Aktifkan perlindungan deregistrasi

Gunakan prosedur berikut untuk mengaktifkan perlindungan deregistrasi.

Console

Untuk mengaktifkan perlindungan deregistrasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Dari bilah filter, pilih Dim iliki oleh saya untuk mencantumkan AMI yang tersedia, atau pilih Gambar dinonaktifkan untuk mencantumkan AMI yang dinonaktifkan.
4. Pilih AMI tempat Anda ingin mengaktifkan perlindungan deregistrasi, lalu pilih Tindakan, Kel ola perlindungan der egi strasi AMI.
5. Di kotak dialog Kelola perlindungan deregistrasi AMI, Anda dapat mengaktifkan perlindungan deregistrasi dengan atau tanpa periode cooldown. Pilih salah satu opsi berikut:
   - Aktifkan dengan periode cooldown 24 jam — Dengan periode cooldown, AMI tidak dapat dibatalkan pendaftaran selama 24 jam ketika perlindungan deregistrasi dimatikan.
   - Aktifkan tanpa cooldown — Tanpa periode cooldown, AMI dapat segera dibatalkan saat perlindungan deregistrasi dimatikan.
6. Pilih Simpan.

AWS CLI

Untuk mengaktifkan perlindungan deregistrasi

Gunakan perintah enable- image-deregistration-protection. Untuk mengaktifkan periode cooldown opsional, sertakan `--with-cooldown` opsi.

```
aws ec2 enable-image-deregistration-protection \
    --image-id ami-0abcdef1234567890 \
    --with-cooldown
```

PowerShell

Untuk mengaktifkan perlindungan deregistrasi

Gunakan Enable-EC2ImageDeregistrationProtection cmdlet. Untuk mengaktifkan periode cooldown opsional, atur `-WithCooldown` parameter `ketrue`.

```
Enable-EC2ImageDeregistrationProtection `
```

```
    -ImageId ami-0abcdef1234567890 `
    -WithCooldown $true
```

### Matikan perlindungan deregistrasi

Gunakan prosedur berikut untuk mematikan perlindungan deregistrasi.

Jika Anda memilih untuk menyertakan periode cooldown 24 jam saat mengaktifkan perlindungan deregistrasi untuk AMI, maka, ketika Anda mematikan perlindungan deregistrasi, Anda tidak akan dapat segera membatalkan pendaftaran AMI. Periode cooldown adalah periode waktu 24 jam di mana perlindungan deregistrasi tetap berlaku bahkan setelah Anda mematikannya. Selama periode cooldown ini, AMI tidak dapat dibatalkan pendaftaran. Setelah periode cooldown berakhir, AMI dapat dideregistrasi.

#### Console

Untuk mematikan perlindungan deregistrasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Dari bilah filter, pilih Dim iliki oleh saya untuk mencantumkan AMI yang tersedia, atau pilih Gambar dinonaktifkan untuk mencantumkan AMI yang dinonaktifkan.
4. Pilih AMI untuk mematikan perlindungan deregistrasi, lalu pilih Tindakan, Kel ola perlindungan der egi strasi AMI.
5. Di kotak dialog Kelola perlindungan deregistrasi AMI, pilih Nonaktifkan.
6. Pilih Simpan.

#### AWS CLI

Untuk mematikan perlindungan deregistrasi

Gunakan perintah disable- image-deregistration-protection.

```
aws ec2 disable-image-deregistration-protection --image-id ami-0abcdef1234567890
```

#### PowerShell

Untuk mematikan perlindungan deregistrasi

Gunakan Disable-EC2ImageDeregistrationProtection cmdlet.

```
Disable-EC2ImageDeregistrationProtection -ImageId ami-0abcdef1234567890
```

# Perilaku peluncuran instans dengan mode boot Amazon EC2

Ketika komputer melakukan boot, perangkat lunak pertama yang berjalan bertanggung jawab untuk menginisialisasi platform dan menyediakan antarmuka bagi sistem operasi untuk melakukan operasi spesifik platform.

Di Amazon EC2, ada dua varian perangkat lunak mode boot yang didukung: Unified Extensible Firmware Interface (UEFI) dan Legacy BIOS.

Parameter mode boot yang mungkin terjadi pada AMI

AMI dapat memiliki salah satu nilai parameter mode boot berikut: `uefi`, `legacy-bios`, atau `uefi-preferred`. Parameter mode boot AMI bersifat opsional. Untuk AMI tanpa parameter mode boot, instans diluncurkan dari AMI tersebut menggunakan nilai mode boot default untuk tipe instans tersebut.

Tujuan parameter mode boot AMI

Parameter mode boot AMI memberi tanda ke Amazon EC2 tentang mode boot mana yang digunakan saat meluncurkan instans. Saat parameter mode boot diatur ke `uefi`, EC2 mencoba untuk meluncurkan instans di UEFI. Jika sistem operasi tidak dikonfigurasi untuk mendukung UEFI, peluncuran instans tidak akan berhasil.

Parameter mode boot UEFI yang disukai

Anda dapat membuat AMI yang mendukung UEFI dan Legacy BIOS dengan menggunakan parameter mode boot `uefi-preferred`. Saat parameter mode boot diatur ke `uefi-preferred`, dan jika tipe instans mendukung UEFI, instans akan diluncurkan di UEFI. Jika tipe instans tidak mendukung UEFI, instans akan diluncurkan di Legacy BIOS.

> **Warning**
>
> Beberapa fitur, seperti UEFI Secure Boot, hanya tersedia pada instans yang di-boot di UEFI. Saat Anda menggunakan parameter mode boot `uefi-preferred` AMI dengan jenis instance yang tidak mendukung UEFI, instance akan diluncurkan sebagai BIOS Legacy

dan UEFI-dependent fitur akan dinonaktifkan. Jika Anda mengandalkan ketersediaan UEFI-dependent fitur, setel parameter mode boot AMI Anda ke`uefi`.

Mode boot default sesuai tipe instans

- Tipe instans Graviton: UEFI
- Tipe instans Intel dan AMD: Legacy BIOS

Dukungan zona

Boot UEFI tidak didukung di Zona Panjang Gelombang.

Topik mode boot

- Persyaratan untuk meluncurkan instance EC2 dalam mode boot UEFI
- Tentukan parameter mode boot dari AMI Amazon EC2
- Tentukan mode boot yang didukung dari jenis instans EC2
- Tentukan mode boot dari instance EC2
- Tentukan mode boot sistem operasi untuk instans EC2 Anda
- Mengatur mode boot dari AMI Amazon EC2
- Variabel UEFI untuk instans Amazon EC2
- Boot Aman UEFI untuk instans Amazon EC2

## Persyaratan untuk meluncurkan instance EC2 dalam mode boot UEFI

Mode boot suatu instance ditentukan oleh konfigurasi AMI, sistem operasi yang terkandung di dalamnya, dan jenis instance. Untuk meluncurkan instance dalam mode boot UEFI, Anda harus memenuhi persyaratan berikut.

AMI

AMI harus dikonfigurasi untuk UEFI sebagai berikut:

- Sistem operasi – sistem operasi yang terdapat dalam AMI harus dikonfigurasi untuk menggunakan UEFI; jika tidak, peluncuran instans akan gagal. Untuk informasi selengkapnya, lihat Tentukan mode boot sistem operasi untuk instans EC2 Anda.

- Parameter mode boot AMI – Parameter mode boot AMI harus diatur ke `uefi` atau `uefi-preferred`. Untuk informasi selengkapnya, lihat Tentukan parameter mode boot dari AMI Amazon EC2.

Linux — AMI Linux berikut mendukung UEFI:

- Amazon Linux 2023
- Amazon Linux 2 (hanya jenis instance Graviton)

Untuk AMI Linux lainnya, Anda harus mengkonfigurasi AMI, mengimpor AMI melalui VM Import/Export, atau mengimpor AMI melalui CloudEndure.

Windows - AMI Windows berikut mendukung UEFI:

- Windows_Server-2025-* (kecuali AMI dengan awalan nama`BIOS-`)
- TPM-Windows_Server-2022-English-Full-Base
- TPM-Windows_Server-2022-English-Core-Base
- TPM-Windows_Server-2019-English-Full-Base
- TPM-Windows_Server-2019-English-Core-Base
- TPM-Windows_Server-2016-English-Full-Base
- TPM-Windows_Server-2016-English-Core-Base

Tipe instans

Semua instans yang dibangun pada Sistem AWS Nitro mendukung UEFI dan BIOS Legacy, kecuali yang berikut ini: instans bare metal, DL1, G4ad, P4, u-3tb1, u-6tb1, u-9tb1, u-12tb1, u-18tb1, u-24tb1, dan VT1. Untuk informasi selengkapnya, lihat the section called “Mode boot tipe instans”.

Tabel berikut menunjukkan bahwa mode boot suatu instans (ditunjukkan oleh kolom Mode boot instans yang dihasilkan) ditentukan oleh kombinasi parameter mode boot AMI (kolom 1), konfigurasi mode boot dari sistem operasi yang berada dalam AMI (kolom 2), dan dukungan mode boot dari tipe instans tersebut (kolom 3).

| Parameter mode boot AMI | Konfigurasi mode boot sistem operasi | Dukungan mode boot tipe instans | Mode boot instans yang dihasilkan |
|---|---|---|---|
| UEFI | UEFI | UEFI | UEFI |

| Parameter mode boot AMI | Konfigurasi mode boot sistem operasi | Dukungan mode boot tipe instans | Mode boot instans yang dihasilkan |
|---|---|---|---|
| Legacy BIOS | Legacy BIOS | Legacy BIOS | Legacy BIOS |
| UEFI Diutamakan | UEFI | UEFI | UEFI |
| UEFI Diutamakan | UEFI | UEFI dan Legacy BIOS | UEFI |
| UEFI Diutamakan | Legacy BIOS | Legacy BIOS | Legacy BIOS |
| UEFI Diutamakan | Legacy BIOS | UEFI dan Legacy BIOS | Legacy BIOS |
| Tidak ada mode boot yang ditentukan - ARM | UEFI | UEFI | UEFI |
| Tidak ada mode boot yang ditentukan - x86 | Legacy BIOS | UEFI dan Legacy BIOS | Legacy BIOS |

## Tentukan parameter mode boot dari AMI Amazon EC2

Parameter mode boot AMI bersifat opsional. AMI dapat memiliki salah satu nilai parameter mode boot berikut: `uefi`, `legacy-bios`, atau `uefi-preferred`.

Beberapa AMI tidak memiliki parameter mode boot. Ketika AMI tidak memiliki parameter mode boot, instans yang diluncurkan dari AMI akan menggunakan nilai default dari tipe instans tersebut, yaitu `uefi` di Graviton, dan `legacy-bios` pada tipe instans Intel dan AMD.

Console

Untuk menentukan parameter mode boot dari AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI, lalu pilih AMI.
3. Periksa bidang Mode boot.

- Nilai uefi menunjukkan bahwa AMI mendukung UEFI.
- Nilai uefi-preferred menunjukkan bahwa AMI mendukung UEFI dan Legacy BIOS.
- Jika tidak ada nilai, instans yang diluncurkan dari AMI menggunakan nilai default dari tipe instans tersebut.

Untuk menentukan parameter mode boot AMI saat meluncurkan instance

Saat meluncurkan sebuah instans menggunakan wizard peluncuran instans, pada langkah untuk memilih AMI, periksa bidang Mode boot. Untuk informasi selengkapnya, lihat Aplikasi dan Gambar OS (Gambar Mesin Amazon).

AWS CLI

Untuk menentukan parameter mode boot dari AMI

Gunakan describe-images perintah untuk menentukan mode boot AMI.

```
aws ec2 describe-images \
    --region us-east-1 \
    --image-id ami-0abcdef1234567890 \
    --query Images[].BootMode \
    --output text
```

Berikut ini adalah output contoh.

```
uefi
```

Dalam output, nilai `uefi` menunjukkan bahwa AMI mendukung UEFI. Nilai `uefi-preferred` menunjukkan bahwa AMI mendukung UEFI dan Legacy BIOS. Jika tidak ada nilai, instans yang diluncurkan dari AMI menggunakan nilai default dari tipe instans tersebut.

PowerShell

Untuk menentukan parameter mode boot dari AMI

Gunakan Get-EC2Image cmdlet untuk menentukan mode boot AMI.

```
Get-EC2Image -Region us-east-1 `
    -ImageId ami-0abcdef1234567890 | Format-List Name, BootMode, TpmSupport
```

Berikut ini adalah output contoh.

```
Name       : TPM-Windows_Server-2016-English-Full-Base-2023.05.10
BootMode   : uefi
TpmSupport : v2.0
```

Dalam output, nilai `BootMode` menunjukkan mode boot AMI. Nilai `uefi` menunjukkan bahwa AMI mendukung UEFI. Nilai `uefi-preferred` menunjukkan bahwa AMI mendukung UEFI dan Legacy BIOS. Jika tidak ada nilai, instans yang diluncurkan dari AMI menggunakan nilai default dari tipe instans tersebut.

## Tentukan mode boot yang didukung dari jenis instans EC2

Anda dapat menentukan mode boot yang didukung dari jenis instance.

Konsol Amazon EC2 tidak menampilkan mode boot yang didukung dari jenis instans.

AWS CLI

Gunakan describe-instance-types perintah untuk menentukan mode boot yang didukung dari jenis instance. `--query`Parameter menyaring output untuk mengembalikan hanya mode boot yang didukung.

Contoh berikut menunjukkan bahwa jenis instance yang ditentukan mendukung mode boot UEFI dan Legacy BIOS.

```
aws ec2 describe-instance-types \
    --instance-types m5.2xlarge \
    --query "InstanceTypes[*].SupportedBootModes"
```

Berikut ini adalah output contoh.

```
[
    [
        "legacy-bios",
        "uefi"
    ]
]
```

Contoh berikut menunjukkan bahwa `t2.xlarge` hanya mendukung Legacy BIOS.

```
aws ec2 describe-instance-types \
    --instance-types t2.xlarge \
    --query "InstanceTypes[*].SupportedBootModes"
```

Berikut ini adalah output contoh.

```
[
    [
        "legacy-bios"
    ]
]
```

PowerShell

Gunakan Get-EC2InstanceType cmdlet untuk menentukan mode boot yang didukung dari jenis instance.

Contoh berikut menunjukkan bahwa `m5.2xlarge` mendukung mode boot UEFI dan Legacy BIOS.

```
Get-EC2InstanceType -InstanceType m5.2xlarge | Format-List InstanceType,
 SupportedBootModes
```

Berikut ini adalah output contoh.

```
InstanceType       : m5.2xlarge
SupportedBootModes : {legacy-bios, uefi}
```

Contoh berikut menunjukkan bahwa `t2.xlarge` hanya mendukung Legacy BIOS.

```
Get-EC2InstanceType -InstanceType t2.xlarge | Format-List InstanceType,
 SupportedBootModes
```

Berikut ini adalah output contoh.

```
InstanceType       : t2.xlarge
SupportedBootModes : {legacy-bios}
```

Untuk menentukan jenis instans yang mendukung UEFI

Anda dapat menentukan jenis instance yang mendukung UEFI. Konsol Amazon EC2 tidak menampilkan dukungan UEFI dari jenis instans.

AWS CLI

Tipe instans yang tersedia berbeda-beda menurut Wilayah AWS. Untuk melihat jenis instance yang tersedia yang mendukung UEFI di suatu Wilayah, gunakan perintah describe-instance-types. Sertakan parameter `--filters` untuk cakupan hasil ke tipe instans yang mendukung UEFI dan parameter `--query` untuk cakupan output ke nilai `InstanceType`.

```
aws ec2 describe-instance-types \
    --filters Name=supported-boot-mode,Values=uefi \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

PowerShell

Tipe instans yang tersedia berbeda-beda menurut Wilayah AWS. Untuk melihat jenis instans yang tersedia yang mendukung UEFI di suatu Wilayah, gunakan Get-EC2InstanceType cmdlet.

```
Get-EC2InstanceType -Filter @{Name="supported-boot-mode"; Values="uefi"} |
 Sort-Object @{E={$_.ProcessorInfo.SupportedArchitectures}}, InstanceType |
 Format-Table InstanceType, SupportedBootModes, `
      @{Name="SupportedArchitectures"; E={$_.ProcessorInfo.SupportedArchitectures}}
```

Untuk menentukan jenis instans yang mendukung UEFI Secure Boot dan mempertahankan variabel non-volatile

Instans bare metal tidak mendukung UEFI Secure Boot dan variabel non-volatile, jadi contoh ini mengecualikannya dari output. Untuk informasi tentang UEFI Secure Boot, lihat Boot Aman UEFI untuk instans Amazon EC2.

AWS CLI

Gunakan perintah describe-instance-types, dan kecualikan instance bare metal dari output.

```
aws ec2 describe-instance-types \
    --filters Name=supported-boot-mode,Values=uefi Name=bare-metal,Values=false \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

PowerShell

Gunakan Get-EC2InstanceType cmdlet, dan kecualikan instance bare metal dari output.

```
Get-EC2InstanceType -Filter `
      @{Name="supported-boot-mode"; Values="uefi"}, `
      @{Name="bare-metal"; Values="false"} |
 Sort-Object @{E={$_.ProcessorInfo.SupportedArchitectures}}, InstanceType |
 Format-Table InstanceType, SupportedBootModes, `
      @{Name="SupportedArchitectures"; E={$_.ProcessorInfo.SupportedArchitectures}}
```

## Tentukan mode boot dari instance EC2

Mode boot sebuah instans ditampilkan di bidang Mode boot di konsol Amazon EC2, dan oleh parameter `currentInstanceBootMode` di AWS CLI.

Apabila sebuah instans diluncurkan, nilai untuk parameter mode boot-nya ditentukan oleh nilai parameter mode boot AMI yang digunakan untuk meluncurkannya, seperti berikut:

- AMI dengan parameter mode boot `uefi` menciptakan sebuah instans dengan parameter `currentInstanceBootMode` `uefi`.
- AMI dengan parameter mode boot `legacy-bios` menciptakan sebuah instans dengan parameter `currentInstanceBootMode` `legacy-bios`.
- AMI dengan parameter mode boot `uefi-preferred` menciptakan instans dengan parameter `currentInstanceBootMode` `uefi` jika tipe instans mendukung UEFI; jika tidak, ia membuat instans dengan parameter `currentInstanceBootMode` `legacy-bios`.
- AMI tanpa nilai parameter mode boot akan menciptakan instans dengan nilai parameter `currentInstanceBootMode` yang bergantung pada apakah arsitektur AMI adalah ARM atau x86 dan mode boot yang didukung tipe instans tersebut. Mode boot default adalah `uefi` pada tipe instans Graviton, dan `legacy-bios` pada tipe instans Intel dan AMD.

Console

Untuk menentukan mode boot suatu instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans, kemudian pilih instans Anda.

3. Di tab Detail, periksa bidang Mode boot.

### AWS CLI

Untuk menentukan mode boot suatu instance

Gunakan perintah describe-instances untuk menentukan mode boot sebuah instans. Anda juga dapat menentukan mode boot AMI yang digunakan untuk membuat instans.

```
aws ec2 describe-instances \
    --region us-east-1 \
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[].Instances[].BootMode \
    --output text
```

Berikut ini adalah output contoh.

```
uefi
```

### PowerShell

Untuk menentukan mode boot suatu instance

Gunakan Get-EC2Image cmdlet untuk menentukan mode boot suatu instance. Anda juga dapat menentukan mode boot AMI yang digunakan untuk membuat instans.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances | Format-List BootMode,
 CurrentInstanceBootMode, InstanceType, ImageId
```

Berikut ini adalah output contoh.

```
BootMode                : uefi
CurrentInstanceBootMode : uefi
InstanceType            : c5a.large
ImageId                 : ami-0abcdef1234567890
```

# Tentukan mode boot sistem operasi untuk instans EC2 Anda

Mode boot AMI memandu Amazon EC2 tentang mode boot apa yang digunakan untuk melakukan boot instans. Untuk melihat apakah sistem operasi instans Anda dikonfigurasi untuk UEFI, Anda perlu menyambung ke instans menggunakan SSH (instans Linux) atau RDP (instans Windows).

Gunakan instruksi untuk sistem operasi instans Anda.

Linux

Untuk menentukan mode boot sistem operasi instans

1. Sambungkan ke instans Linux Anda menggunakan SSH.
2. Untuk melihat mode boot sistem operasi, coba salah satu hal berikut ini:

   - Jalankan perintah berikut.

     ```
     [ec2-user ~]$ sudo /usr/sbin/efibootmgr
     ```

     Output yang diharapkan dari sebuah instans yang diboot dalam mode boot UEFI

     ```
     BootCurrent: 0001
     Timeout: 0 seconds
     BootOrder: 0000,0001
     Boot0000* UiApp
     Boot0001* UEFI Amazon Elastic Block Store vol-xyz
     ```

   - Jalankan perintah berikut untuk memverifikasi keberadaan direktori `/sys/firmware/efi`. Direktori ini hanya ada jika instans boot menggunakan UEFI. Jika direktori ini tidak ada, perintah akan menampilkan `Legacy BIOS Boot Detected`.

     ```
     [ec2-user ~]$ [ -d /sys/firmware/efi ] && echo "UEFI Boot Detected" || echo
      "Legacy BIOS Boot Detected"
     ```

     Output yang diharapkan dari sebuah instans yang diboot dalam mode boot UEFI

     ```
     UEFI Boot Detected
     ```

     Output yang diharapkan dari instans yang boot dalam mode boot Legacy BIOS

```
Legacy BIOS Boot Detected
```

- Jalankan perintah berikut untuk memverifikasi bahwa EFI muncul di output `dmesg`.

```
[ec2-user ~]$ dmesg | grep -i "EFI"
```

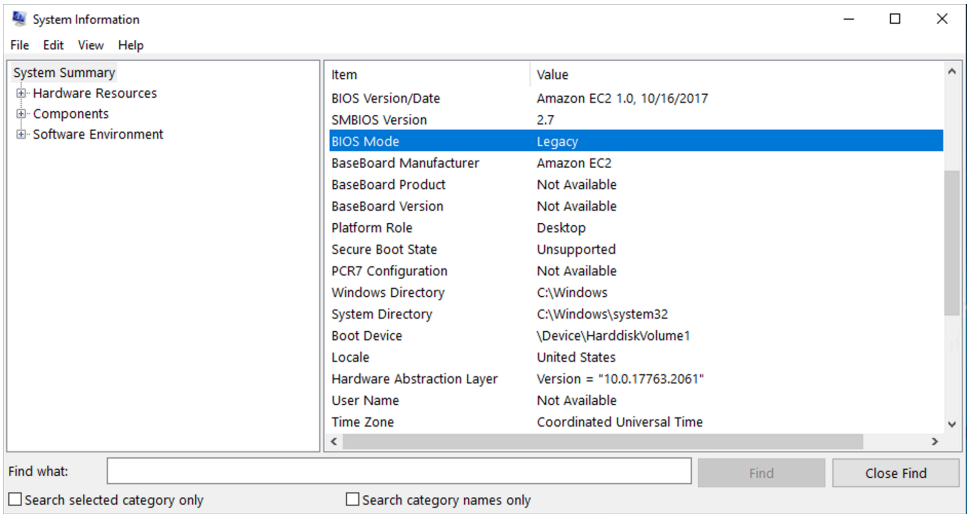
Output yang diharapkan dari sebuah instans yang diboot dalam mode boot UEFI

```
[    0.000000] efi: Getting EFI parameters from FDT:
[    0.000000] efi: EFI v2.70 by EDK II
```

Windows

Untuk menentukan mode boot sistem operasi instans

1. Sambungkan ke instans Windows Anda menggunakan RDP.
2. Pergi ke Informasi Sistem dan periksa baris Mode BIOS.




## Mengatur mode boot dari AMI Amazon EC2

Secara default, AMI mewarisi mode boot dari instance EC2 yang digunakan untuk membuat AMI. Misalnya, jika Anda membuat AMI dari instance EC2 yang berjalan di BIOS Legacy, mode boot AMI baru adalah`legacy-bios`. Jika Anda membuat AMI dari instance EC2 dengan mode boot`uefi-preferred`, mode boot AMI baru adalah`uefi-preferred`.

Saat Anda mendaftarkan AMI, Anda dapat mengatur mode boot AMI keuefi,legacy-bios, atauuefi-preferred.

Ketika mode boot AMI diatur ke uefi-preferred, instans akan melakukan boot sebagai berikut:

- Untuk tipe instans yang mendukung UEFI dan Legacy BIOS (misalnya, m5.large), instans boot menggunakan UEFI.
- Untuk tipe instans yang hanya mendukung Legacy BIOS (misalnya, m4.large), instans boot menggunakan Legacy BIOS.

Jika Anda mengatur mode boot AMI ke uefi-preferred, sistem operasi harus mendukung kemampuan untuk melakukan boot UEFI dan Legacy BIOS.

Untuk mengonversi BIOS-based instance Legacy yang ada ke UEFI, atau UEFI-based instance yang ada ke BIOS Legacy, Anda harus terlebih dahulu memodifikasi volume instans dan sistem operasi untuk mendukung mode boot yang dipilih. Kemudian, buat snapshot volume. Terakhir, buat AMI dari snapshot.

Pertimbangan-pertimbangan

- Menetapkan parameter mode boot AMI tidak secara otomatis mengonfigurasi sistem operasi untuk mode boot tersebut. Anda harus terlebih dahulu membuat modifikasi yang sesuai untuk volume instans dan sistem operasi untuk mendukung booting menggunakan mode boot yang dipilih. Jika tidak, AMI yang dihasilkan tidak dapat digunakan. Misalnya, jika Anda mengonversi instance BIOS-based Windows Legacy ke UEFI, Anda dapat menggunakan alat MBR2GPT dari Microsoft untuk mengonversi disk sistem dari MBR ke GPT. Perubahan yang diperlukan adalah perubahan khusus sistem operasi. Untuk informasi lebih lanjut, lihat manual untuk sistem operasi Anda.
- Anda tidak dapat menggunakan register-image perintah atau Register-EC2Image cmdlet untuk membuat AMI yang mendukung NitroTPM dan UEFI Preferred.
- Beberapa fitur, seperti UEFI Secure Boot, hanya tersedia pada instans yang di-boot di UEFI. Saat Anda menggunakan parameter mode boot uefi-preferred AMI dengan jenis instance yang tidak mendukung UEFI, instance diluncurkan sebagai BIOS Legacy dan UEFI-dependent fitur tersebut dinonaktifkan. Jika Anda mengandalkan ketersediaan UEFI-dependent fitur, setel parameter mode boot AMI Anda keuefi.

AWS CLI

Untuk mengatur mode boot AMI

1. Buat modifikasi yang sesuai pada volume instans dan sistem operasi untuk mendukung booting dengan menggunakan mode boot yang dipilih. Perubahan yang diperlukan adalah perubahan khusus sistem operasi. Untuk informasi lebih lanjut, lihat manual untuk sistem operasi Anda.

   > **Warning**
   >
   > Jika Anda tidak melakukan langkah ini, AMI tidak akan dapat digunakan.

2. Untuk menemukan ID volume instans, gunakan perintah describe-instances. Anda akan membuat snapshot volume ini di langkah berikutnya.

   ```
   aws ec2 describe-instances \
       --instance-ids i-1234567890abcdef0 \
       --query Reservations[].Instances[].BlockDeviceMappings
   ```

   Berikut ini adalah output contoh.

   ```
   [
       [
           {
               "DeviceName": "/dev/xvda",
               "Ebs": {
                   "AttachTime": "2024-07-11T01:05:51+00:00",
                   "DeleteOnTermination": true,
                   "Status": "attached",
                   "VolumeId": "vol-1234567890abcdef0"
               }
           }
       ]
   ]
   ```

3. Untuk membuat snapshot volume, gunakan perintah create-snapshot. Gunakan ID volume dari langkah sebelumnya.

   ```
   aws ec2 create-snapshot \
       --volume-id vol-01234567890abcdef \
   ```

```
    --description "my snapshot"
```

Berikut ini adalah output contoh.

```
{
    "Description": "my snapshot",
    "Encrypted": false,
    "OwnerId": "123456789012",
    "Progress": "",
    "SnapshotId": "snap-0abcdef1234567890",
    "StartTime": "",
    "State": "pending",
    "VolumeId": "vol-01234567890abcdef",
    "VolumeSize": 30,
    "Tags": []
}
```

4. Tunggu sampai keadaan snapshot `completed` sebelum Anda pergi ke langkah berikutnya. Untuk mendapatkan status snapshot, gunakan describe-snapshots perintah dengan ID snapshot dari langkah sebelumnya.

```
aws ec2 describe-snapshots \
    --snapshot-ids snap-0abcdef1234567890 \
    --query Snapshots[].State \
    --output text
```

Berikut ini adalah output contoh.

```
completed
```

5. Untuk membuat AMI baru, gunakan perintah register-image. Gunakan nilai `SnapshotId` dari output dariCreateSnapshot.

   - Untuk mengatur mode boot ke UEFI, tambahkan `--boot-mode` parameter dengan nilai`uefi`.

```
aws ec2 register-image \
   --description "my image" \
   --name "my-image" \
   --block-device-mappings "DeviceName=/dev/
sda1,Ebs={SnapshotId=snap-0abcdef1234567890,DeleteOnTermination=true}" \
```

```
    --root-device-name /dev/sda1 \
    --virtualization-type hvm \
    --ena-support \
    --boot-mode uefi
```

- Untuk mengatur mode boot keuefi-preferred, atur nilai `--boot-mode` ke `uefi-preferred`

```
aws ec2 register-image \
    --description "my description" \
    --name "my-image" \
    --block-device-mappings "DeviceName=/dev/
sda1,Ebs={SnapshotId=snap-0abcdef1234567890,DeleteOnTermination=true}" \
    --root-device-name /dev/sda1 \
    --virtualization-type hvm \
    --ena-support \
    --boot-mode uefi-preferred
```

6. (Opsional) Untuk memverifikasi bahwa AMI yang baru dibuat memiliki mode boot yang Anda tentukan, gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-id ami-1234567890abcdef0 \
    --query Images[].BootMode \
    --output text
```

Berikut ini adalah output contoh.

```
uefi
```

PowerShell

Untuk mengatur mode boot AMI

1. Buat modifikasi yang sesuai pada volume instans dan sistem operasi untuk mendukung booting dengan menggunakan mode boot yang dipilih. Perubahan yang diperlukan adalah perubahan khusus sistem operasi. Untuk informasi lebih lanjut, lihat manual untuk sistem operasi Anda.

> **Warning**
>
> Jika Anda tidak melakukan langkah ini, AMI tidak akan dapat digunakan.

2. Untuk menemukan ID volume instance, gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances.BlockDeviceMappings.Ebs
```

Berikut ini adalah output contoh.

```
AssociatedResource  :
AttachTime          : 7/11/2024 1:05:51 AM
DeleteOnTermination : True
Operator            :
Status              : attached
VolumeId            : vol-01234567890abcdef
```

3. Untuk membuat snapshot volume, gunakan New-EC2Snapshot cmdlet. Gunakan ID volume dari langkah sebelumnya.

```
New-EC2Snapshot `
    -VolumeId vol-01234567890abcdef `
    -Description "my snapshot"
```

Berikut ini adalah output contoh.

```
AvailabilityZone          :
Description               : my snapshot
Encrypted                 : False
FullSnapshotSizeInBytes   : 0
KmsKeyId                  :
OwnerId                   : 123456789012
RestoreExpiryTime         :
SnapshotId                : snap-0abcdef1234567890
SseType                   :
StartTime                 : 4/25/2025 6:08:59 PM
State                     : pending
StateMessage              :
VolumeId                  : vol-01234567890abcdef
```

```
VolumeSize                : 30
```

4. Tunggu sampai keadaan snapshot `completed` sebelum Anda pergi ke langkah berikutnya. Untuk mendapatkan status snapshot, gunakan Get-EC2Snapshot cmdlet dengan ID snapshot dari langkah sebelumnya.

```
(Get-EC2Snapshot `
    -SnapshotId snap-0abcdef1234567890).State.Value
```

Berikut ini adalah output contoh.

```
completed
```

5. Untuk membuat AMI baru, gunakan Register-EC2Image cmdlet. Gunakan nilai `SnapshotId` dari output dariNew-EC2Snapshot.

- Untuk mengatur mode boot ke UEFI, tambahkan `-BootMode` parameter dengan nilai`uefi`.

```
$block = @{SnapshotId=snap-0abcdef1234567890}
Register-EC2Image `
   -Description "my image" `
   -Name "my-image" `
   -BlockDeviceMapping @{DeviceName="/dev/xvda";Ebs=$block} `
   -RootDeviceName /dev/xvda `
   -EnaSupport $true `
   -BootMode uefi
```

- Untuk mengatur mode boot ke`uefi-preferred`, atur nilai `-BootMode` ke `uefi-preferred`

```
$block = @{SnapshotId=snap-0abcdef1234567890}
Register-EC2Image `
   -Description "my image" `
   -Name "my-image" `
   -BlockDeviceMapping @{DeviceName="/dev/xvda";Ebs=$block} `
   -RootDeviceName /dev/xvda `
   -EnaSupport $true `
   -BootMode uefi-preferred
```

6. (Opsional) Untuk memverifikasi bahwa AMI yang baru dibuat memiliki mode boot yang Anda tentukan, gunakan cmdlet. Get-EC2Image

```
(Get-EC2Image `
    -ImageId ami-1234567890abcdef0).BootMode.Value
```

Berikut ini adalah output contoh.

```
uefi
```

## Variabel UEFI untuk instans Amazon EC2

Saat Anda meluncurkan instans di mana mode boot diatur ke UEFI, penyimpanan nilai kunci untuk variabel akan dibuat. Penyimpanan dapat digunakan oleh UEFI dan sistem operasi instans untuk menyimpan variabel UEFI.

Variabel UEFI digunakan oleh boot loader dan sistem operasi untuk mengonfigurasi startup sistem awal. Variabel ini memungkinkan sistem operasi untuk mengelola pengaturan tertentu dari proses boot, seperti urutan boot, atau mengelola kunci untuk UEFI Secure Boot.

> **Warning**
>
> Siapa pun yang dapat terhubung ke instance (dan kemungkinan perangkat lunak apa pun yang berjalan pada instance), atau siapa pun yang memiliki izin untuk menggunakan GetInstanceUefiData API pada instance dapat membaca variabel. Anda tidak boleh menyimpan data sensitif, seperti sandi atau informasi identitas pribadi, di penyimpanan variabel UEFI.

Persistensi variabel UEFI

- Untuk instans yang diluncurkan pada atau sebelum 10 Mei 2022, variabel UEFI dihapus saat boot ulang atau berhenti.
- Untuk contoh yang diluncurkan pada atau setelah 11 Mei 2022, variabel UEFI yang ditandai sebagai non-volatile dipertahankan saat reboot dan. stop/start
- Instans bare metal tidak mempertahankan variabel non-volatile UEFI di seluruh operasi instance stop/start.

# Boot Aman UEFI untuk instans Amazon EC2

UEFI Secure Boot dibangun di atas proses boot aman Amazon EC2, dan menyediakan pertahanan mendalam tambahan yang membantu pelanggan mengamankan perangkat lunak dari ancaman selama boot ulang. UEFI Secure Boot memastikan bahwa instans hanya melakukan boot perangkat lunak yang diberi tanda dengan kunci kriptografi. Kunci disimpan dalam basis data kunci di penyimpanan variabel non-volatile UEFI. UEFI Secure Boot mencegah modifikasi yang tidak sah dari aliran boot instans.

Daftar Isi


- Cara kerja UEFI Secure Boot dengan instans Amazon EC2
- Persyaratan untuk Boot Aman UEFI di Amazon EC2
- Memverifikasi apakah instans Amazon EC2 diaktifkan untuk UEFI Secure Boot
- Buat AMI Linux dengan kunci Boot Aman UEFI khusus
- Buat AWS blob biner untuk UEFI Secure Boot


## Cara kerja UEFI Secure Boot dengan instans Amazon EC2

UEFI Secure Boot adalah fitur yang ditentukan dalam UEFI, yang menyediakan verifikasi tentang keadaan rantai boot. UEFI Secure Boot dirancang untuk memastikan bahwa hanya binari UEFI yang terverifikasi secara kriptografis yang akan dieksekusi setelah inisialisasi mandiri pada firmware. Binari ini termasuk driver UEFI dan bootloader utama, serta komponen yang dimuat rantai.

UEFI Secure Boot menetapkan empat basis data utama, yang digunakan dalam rantai kepercayaan. Basis data disimpan di penyimpanan variabel UEFI.

Rantai kepercayaan tersebut adalah sebagai berikut:

Basis data kunci platform (PK)

> Basis data PK adalah root kepercayaan. Basis data ini berisi satu kunci PK publik yang digunakan dalam rantai kepercayaan untuk memperbarui basis data kunci untuk pertukaran kunci (KEK).
>
> Untuk mengubah basis data PK, Anda harus memiliki kunci PK privat untuk menandatangani permintaan pembaruan. Ini termasuk menghapus basis data PK dengan menulis kunci PK kosong.

Basis data kunci untuk pertukaran kunci (KEK)

Basis data KEK adalah daftar kunci KEK publik yang digunakan dalam rantai kepercayaan untuk memperbarui basis data tanda tangan (db) dan denylist (dbx).

Untuk mengubah basis data KEK publik, Anda harus memiliki kunci PK privat untuk menandatangani permintaan pembaruan.

Basis data tanda tangan (db)

Basis data db adalah daftar kunci publik dan hash yang digunakan dalam rantai kepercayaan untuk memvalidasi semua binari boot UEFI.

Untuk mengubah basis data db, Anda harus memiliki kunci PK privat atau salah satu kunci KEK privat untuk menandatangani permintaan pembaruan.

Basis data denylist tanda tangan (dbx)

Basis data dbx adalah daftar kunci publik dan hash biner yang tidak tepercaya, dan digunakan dalam rantai kepercayaan sebagai file pencabutan.

Basis data dbx selalu diutamakan daripada semua basis data kunci lainnya.

Untuk mengubah basis data dbx, Anda harus memiliki kunci PK privat atau kunci KEK privat apa pun untuk menandatangani permintaan pembaruan.

Forum UEFI mempertahankan dbx yang tersedia untuk umum untuk banyak binari dan sertifikat buruk yang diketahui di. https://uefi.org/revocationlistfile

> **Important**
>
> UEFI Secure Boot memberlakukan validasi tanda tangan pada binari UEFI apa pun. Untuk mengizinkan eksekusi biner UEFI di UEFI Secure Boot, Anda menandatanganinya dengan salah satu kunci db privat yang dijelaskan di atas.

Secara default, UEFI Secure Boot dinonaktifkan dan sistem ada pada `SetupMode`. Ketika sistem ada di `SetupMode`, semua variabel kunci dapat diperbarui tanpa tanda tangan kriptografis. Ketika PK diatur, UEFI Secure Boot diaktifkan dan SetupMode keluar.

# Persyaratan untuk Boot Aman UEFI di Amazon EC2

Saat Anda meluncurkan instans Amazon EC2 dengan AMI yang didukung dan jenis instans yang didukung, instans tersebut akan secara otomatis memvalidasi binari boot UEFI terhadap database Boot Aman UEFI miliknya. Tidak diperlukan konfigurasi tambahan. Anda juga dapat mengonfigurasi UEFI Secure Boot pada sebuah instans setelah diluncurkan.

> **Note**
>
> UEFI Secure Boot melindungi instans Anda dan sistem operasinya dari perubahan aliran boot. Jika Anda membuat AMI baru dari AMI sumber yang mengaktifkan UEFI Secure Boot dan memodifikasi parameter tertentu selama proses penyalinan, seperti mengubah di `UefiData` dalam AMI, Anda dapat menonaktifkan UEFI Secure Boot.

Daftar Isi


- AMI yang didukung
- Tipe instans yang didukung


## AMI yang didukung

### AMI Linux

Untuk meluncurkan instance Linux, AMI Linux harus mengaktifkan UEFI Secure Boot.

Amazon Linux mendukung UEFI Secure Boot dimulai dengan AL2023 versi 2023.1. Namun, UEFI Secure Boot tidak diaktifkan di AMI default. Untuk informasi selengkapnya, lihat UEFI Secure Boot di Panduan Pengguna AL2023. Versi lama AMI Amazon Linux tidak diaktifkan untuk UEFI Secure Boot. Agar dapat menggunakan AMI yang didukung, Anda harus melakukan sejumlah langkah konfigurasi pada Linux AMI Anda sendiri. Untuk informasi selengkapnya, lihat Buat AMI Linux dengan kunci Boot Aman UEFI khusus.

### AMI Windows

Untuk meluncurkan instance Windows, AMI Windows harus mengaktifkan UEFI Secure Boot. Untuk menemukan AMI AWS Windows yang telah dikonfigurasi sebelumnya untuk UEFI Secure Boot dengan kunci Microsoft, lihat M enemukan AMI Windows Server yang dikonfigurasi dengan NitroTPM dan UEFI Secure Boot di Referensi AMI Windows. AWS

Saat ini, kami tidak mendukung impor Windows dengan UEFI Secure Boot dengan menggunakan perintah import-image.

Tipe instans yang didukung

Semua jenis instans virtual yang mendukung UEFI juga mendukung UEFI Secure Boot. Untuk tipe instans yang mendukung UEFI Secure Boot, lihat Persyaratan untuk mode boot UEFI.

> **Note**
>
> Tipe instans bare metal tidak mendukung UEFI Secure Boot.

## Memverifikasi apakah instans Amazon EC2 diaktifkan untuk UEFI Secure Boot

Anda dapat menggunakan prosedur berikut untuk menentukan apakah Amazon EC2 diaktifkan untuk UEFI Secure Boot.

Instans Linux

Anda dapat menggunakan utilitas `mokutil` untuk memverifikasi apakah instans Linux diaktifkan untuk UEFI Secure Boot. Jika `mokutil` tidak diinstal pada instans Anda, Anda harus menginstalnya. Untuk petunjuk penginstalan untuk Amazon Linux 2, lihat M enemukan dan instal paket perangkat lunak pada instans Amazon Linux 2. Untuk distribusi Linux lainnya, lihat dokumentasi spesifiknya.

Untuk memverifikasi apakah sebuah instans Linux diaktifkan untuk UEFI Secure Boot

Hubungkan ke instance Anda dan jalankan perintah berikut seperti `root` di jendela terminal.

```
mokutil --sb-state
```

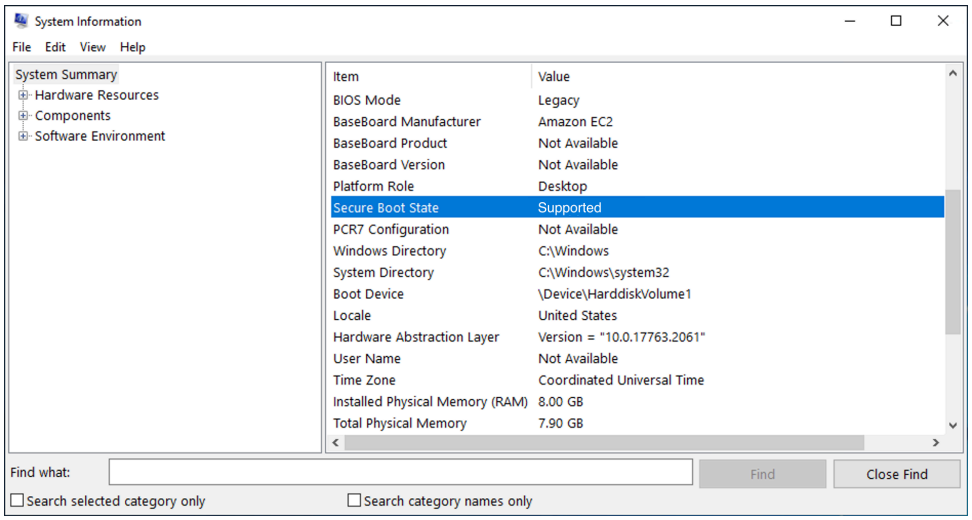
Berikut ini adalah output contoh.

- Jika UEFI Secure Boot diaktifkan, output berisi `SecureBoot enabled`.
- Jika UEFI Secure Boot tidak diaktifkan, output berisi `SecureBoot disabled` atau `Failed to read SecureBoot`.

Instans Windows

Untuk memverifikasi apakah sebuah instans Windows diaktifkan untuk UEFI Secure Boot

1. Terhubung ke instans Anda.
2. Buka alat msinfo32.
3. Periksa bidang Kondisi Secure Boot. Jika UEFI Secure Boot diaktifkan, nilainya Di dukung, seperti yang ditunjukkan pada gambar berikut.




Anda juga dapat menggunakan PowerShell Cmdlet Windows `Confirm-SecureBootUEFI` untuk memeriksa status Boot Aman. Untuk informasi selengkapnya tentang cmdlet, lihat Confirm-SecureBootUEFI di Dokumentasi Microsoft.

## Buat AMI Linux dengan kunci Boot Aman UEFI khusus

Instruksi ini menunjukkan cara membuat AMI Linux dengan UEFI Secure Boot dan kunci pribadi yang dibuat khusus. Amazon Linux mendukung UEFI Secure Boot dimulai dengan AL2023 versi 2023.1. Untuk informasi selengkapnya, lihat UEFI Secure Boot di AL2023 di Panduan Pengguna Amazon Linux 2023.

> **Important**
>
> Prosedur berikut ini ditujukan untuk pengguna tingkat lanjut saja. Anda harus memiliki pengetahuan yang cukup tentang alur boot distribusi SSL dan Linux untuk menggunakan prosedur ini.

Prasyarat

- Alat-alat berikut akan digunakan:
  - OpenSSL — https://www.openssl.org/
  - efivar — https://github.com/rhboot/efivar
  - efitools — https://git.kernel.org/pub/scm/linux/kernel/git/jejb/efitools.git/
  - perintah get-instance-uefi-data
- Instans Linux Anda harus telah diluncurkan dengan AMI Linux yang mendukung mode boot UEFI, dan memiliki data non-volatile.

Instans yang baru dibuat tanpa kunci UEFI Secure Boot akan dibuat di `SetupMode`, yang memungkinkan Anda untuk mendaftarkan kunci Anda sendiri. Beberapa AMI telah dikonfigurasi sebelumnya dengan UEFI Secure Boot dan Anda tidak dapat mengubah kunci yang ada. Jika Anda ingin mengubah kunci, Anda harus membuat AMI baru berdasarkan AMI yang asli.

Anda memiliki dua cara untuk menyebarkan kunci di penyimpanan variabel, yang dijelaskan dalam Opsi A dan Opsi B di bawah ini. Opsi A menjelaskan bagaimana melakukan ini dari dalam instans, meniru aliran perangkat keras nyata. Opsi B menjelaskan cara membuat gumpalan biner, yang kemudian diteruskan sebagai file base64 saat Anda membuat AMI. Untuk kedua opsi, Anda harus terlebih dahulu membuat tiga pasang kunci, yang digunakan untuk rantai kepercayaan.

Untuk membuat AMI Linux yang mendukung UEFI Secure Boot, pertama buat tiga pasangan kunci, lalu selesaikan Opsi A atau Opsi B, tetapi tidak keduanya:

- Tugas 1: Buat pasangan kunci
- Tugas 2 - Opsi A: Tambahkan kunci ke penyimpanan variabel dari dalam instance
- Tugas 2 - Opsi B: Buat gumpalan biner yang berisi penyimpanan variabel yang telah diisi sebelumnya

Tugas 1: Buat pasangan kunci

UEFI Secure Boot didasarkan pada tiga basis data utama berikut, yang digunakan dalam rantai kepercayaan: kunci platform (PK), kunci untuk pertukaran kunci (KEK), dan basis data (db) tanda tangan.[1]

Anda membuat setiap kunci pada instans. Untuk menyiapkan kunci publik dalam format yang valid untuk standar UEFI Secure Boot, Anda membuat sertifikat untuk setiap kunci. `DER` mendefinisikan

format SSL (pengodean biner suatu format). Anda kemudian mengonversi setiap sertifikat menjadi daftar tanda tangan UEFI, yang merupakan format biner yang dipahami oleh UEFI Secure Boot. Terakhir, Anda menandatangani setiap sertifikat dengan kunci yang relevan.

Tugas

- Bersiap untuk membuat pasangan kunci
- Pasangan kunci 1: Buat kunci platform (PK)
- Pasangan kunci 2: Buat kunci untuk pertukaran kunci (KEK)
- Pasangan kunci 3: Buat basis data (db) tanda tangan
- Tanda tangani gambar boot (kernel) dengan kunci privat

### Bersiap untuk membuat pasangan kunci

Sebelum membuat pasangan kunci, buat pengidentifikasi unik global (GUID) untuk digunakan dalam pembuatan kunci.

1. Hubungkan ke instans.
2. Jalankan perintah berikut di prompt shell.

```
uuidgen --random > GUID.txt
```

### Pasangan kunci 1: Buat kunci platform (PK)

PK adalah root kepercayaan untuk instans UEFI Secure Boot. PK privat digunakan untuk memperbarui KEK, yang nantinya dapat digunakan untuk menambahkan kunci resmi ke basis data (db) tanda tangan.

X.509 Standar ini digunakan untuk membuat pasangan kunci. Untuk informasi tentang standar, lihat X.509 di Wikipedia.

Untuk membuat PK

1. Buat kunci. Anda harus memberi nama variabel PK.

```
openssl req -newkey rsa:4096 -nodes -keyout PK.key -new -x509 -sha256 -days 3650 -
subj "/CN=Platform key/" -out PK.crt
```

Parameter berikut ditentukan:

- `-keyout PK.key` – File kunci privat.
- `-days 3650` – Jumlah hari sertifikat tersebut valid.
- `-out PK.crt` – Sertifikat yang digunakan untuk membuat variabel UEFI.
- `CN=`*`Platform key`* – Nama umum (CN) untuk kunci. Anda dapat memasukkan nama organisasi Anda sendiri sebagai gantinya*`Platform key`*.

2. Buat sertifikat.

```
openssl x509 -outform DER -in PK.crt -out PK.cer
```

3. Konversi sertifikat menjadi daftar tanda tangan UEFI.

```
cert-to-efi-sig-list -g "$(< GUID.txt)" PK.crt PK.esl
```

4. Tanda tangani daftar tanda tangan UEFI dengan PK privat (yang ditandatangani sendiri).

```
sign-efi-sig-list -g "$(< GUID.txt)" -k PK.key -c PK.crt PK PK.esl PK.auth
```

Pasangan kunci 2: Buat kunci untuk pertukaran kunci (KEK)

KEK privat digunakan untuk menambahkan kunci ke db, yang merupakan daftar tanda tangan resmi untuk boot pada sistem.

Untuk membuat PK

1. Buat kunci.

```
openssl req -newkey rsa:4096 -nodes -keyout KEK.key -new -x509 -sha256 -days 3650 -
subj "/CN=Key Exchange Key/" -out KEK.crt
```

2. Buat sertifikat.

```
openssl x509 -outform DER -in KEK.crt -out KEK.cer
```

3. Konversi sertifikat menjadi daftar tanda tangan UEFI.

```
cert-to-efi-sig-list -g "$(< GUID.txt)" KEK.crt KEK.esl
```

4. Tanda tangani daftar tanda tangan dengan PK privat.

```
sign-efi-sig-list -g "$(< GUID.txt)" -k PK.key -c PK.crt KEK KEK.esl KEK.auth
```

Pasangan kunci 3: Buat basis data (db) tanda tangan

Daftar db berisi kunci resmi yang diizinkan untuk di-boot pada sistem. Untuk memodifikasi daftar ini, diperlukan KEK privat. Gambar boot akan ditandatangani dengan kunci privat yang dibuat pada langkah ini.

Untuk membuat PK

1. Buat kunci.

```
openssl req -newkey rsa:4096 -nodes -keyout db.key -new -x509 -sha256 -days 3650 -
subj "/CN=Signature Database key/" -out db.crt
```

2. Buat sertifikat.

```
openssl x509 -outform DER -in db.crt -out db.cer
```

3. Konversi sertifikat menjadi daftar tanda tangan UEFI.

```
cert-to-efi-sig-list -g "$(< GUID.txt)" db.crt db.esl
```

4. Tanda tangani daftar tanda tangan dengan KEK privat.

```
sign-efi-sig-list -g "$(< GUID.txt)" -k KEK.key -c KEK.crt db db.esl db.auth
```

Tanda tangani gambar boot (kernel) dengan kunci privat

Untuk Ubuntu 22.04, gambar berikut memerlukan tanda tangan.

```
/boot/efi/EFI/ubuntu/shimx64.efi
/boot/efi/EFI/ubuntu/mmx64.efi
/boot/efi/EFI/ubuntu/grubx64.efi
/boot/vmlinuz
```

Untuk menandatangani gambar

Gunakan sintaksis berikut untuk menandatangani gambar.

```
sbsign --key db.key --cert db.crt --output /boot/vmlinuz /boot/vmlinuz
```

> **Note**
>
> Anda harus menandatangani semua kernel baru. */boot/vmlinuz* biasanya akan symlink ke kernel yang terakhir diinstal.

Lihat dokumentasi distribusi Anda untuk menemukan rantai boot dan gambar yang diperlukan.

[1] Terima kasih kepada ArchWiki komunitas atas semua pekerjaan yang telah mereka lakukan. Perintah untuk membuat PK, membuat KEK, membuat DB, dan menandatangani gambar berasal dari Creating keys, yang ditulis oleh Tim Pem ArchWiki eliharaan kontributor and/or . ArchWiki

Tugas 2 - Opsi A: Tambahkan kunci ke penyimpanan variabel dari dalam instance

Setelah Anda membuat tiga pasang kunci, Anda dapat terhubung ke instans Anda dan menambahkan kunci ke penyimpanan variabel dari dalam instans dengan menyelesaikan langkah-langkah berikut. Atau, selesaikan langkah-langkah dithe section called “Tugas 2 - Opsi B: Buat gumpalan biner yang berisi penyimpanan variabel yang telah diisi sebelumnya”.

Langkah-langkah Opsi A:

- Langkah 1: Luncurkan instans yang akan mendukung UEFI Secure Boot
- Langkah 2: Konfigurasikan instans untuk mendukung UEFI Secure Boot
- Langkah 3: Buat AMI dari instans

Langkah 1: Luncurkan instans yang akan mendukung UEFI Secure Boot

Ketika Anda meluncurkan sebuah instans dengan prasyarat berikut, instans kemudian akan siap untuk dikonfigurasi untuk mendukung UEFI Secure Boot. Anda hanya dapat mengaktifkan dukungan untuk UEFI Secure Boot pada instans saat peluncuran; Anda tidak dapat mengaktifkannya nanti.

Prasyarat

- AMI – AMI Linux harus mendukung mode boot UEFI. Untuk memverifikasi bahwa AMI mendukung mode boot UEFI, parameter mode boot AMI harus uefi. Untuk informasi selengkapnya, lihat Tentukan parameter mode boot dari AMI Amazon EC2.

Perhatikan bahwa AWS hanya menyediakan AMI Linux yang dikonfigurasi untuk mendukung UEFI untuk jenis Graviton-based contoh. AWS saat ini tidak menyediakan x86_64 Linux AMI yang mendukung mode boot UEFI. Anda dapat mengonfigurasi AMI Anda sendiri untuk mendukung mode boot UEFI untuk semua arsitektur. Untuk mengonfigurasi AMI Anda sendiri untuk mendukung mode boot UEFI, Anda harus melakukan sejumlah langkah konfigurasi pada AMI Anda sendiri. Untuk informasi selengkapnya, lihat Mengatur mode boot dari AMI Amazon EC2.

- Tipe instans – Semua tipe instans virtual yang mendukung UEFI juga mendukung UEFI Secure Boot. Tipe instans bare metal tidak mendukung UEFI Secure Boot. Untuk tipe instans yang mendukung UEFI Secure Boot, lihat Persyaratan untuk mode boot UEFI.
- Luncurkan instans Anda setelah rilis UEFI Secure Boot. Hanya instans yang diluncurkan setelah 10 Mei 2022 (saat UEFI Secure Boot dirilis) yang dapat mendukung UEFI Secure Boot.

Setelah Anda meluncurkan instans Anda, Anda dapat memverifikasi bahwa instans siap dikonfigurasi untuk mendukung UEFI Secure Boot (dengan kata lain, Anda dapat melanjutkan ke Langkah 2) dengan memeriksa apakah tersedia data UEFI. Keberadaan data UEFI menunjukkan bahwa data non-volatile tetap ada.

Untuk memverifikasi apakah instans Anda siap untuk Langkah 2

Gunakan perintah get-instance-uefi-data dan tentukan ID instans.

```
aws ec2 get-instance-uefi-data --instance-id i-1234567890abcdef0
```

Instans siap untuk Langkah 2 jika data UEFI sudah tersedia dalam output. Jika output kosong, instans tidak dapat dikonfigurasi untuk mendukung UEFI Secure Boot. Hal ini dapat terjadi jika instans Anda diluncurkan sebelum dukungan UEFI Secure Boot tersedia. Luncurkan instans baru dan coba lagi.

Langkah 2: Konfigurasikan instans untuk mendukung UEFI Secure Boot

Daftarkan pasangan kunci di penyimpanan variabel UEFI Anda pada instans

> **Warning**
>
> Anda harus menandatangani gambar boot Anda setelah Anda mendaftarkan kunci, jika tidak, Anda tidak akan dapat melakukan boot instans Anda.

Setelah Anda membuat daftar tanda tangan UEFI yang ditandatangani (PK, KEK, dan db), tanda tangan tersebut harus terdaftar ke firmware UEFI.

Penulisan ke variabel PK hanya dapat dilakukan jika:

- Belum ada PK yang terdaftar, yang ditunjukkan jika variabel `SetupMode` nya `1`. Periksa ini dengan menggunakan perintah berikut. Outputnya adalah `1` atau `0`.

  ```
  efivar -d -n 8be4df61-93ca-11d2-aa0d-00e098032b8c-SetupMode
  ```

- PK yang baru ditandatangani oleh kunci privat dari PK yang ada.

Untuk mendaftarkan kunci di penyimpanan variabel UEFI Anda

Perintah berikut harus dijalankan pada instans.

Jika SetupMode diaktifkan (nilainya1), kunci dapat didaftarkan dengan menjalankan perintah berikut pada instance:

```
[ec2-user ~]$ efi-updatevar -f db.auth db
```

```
[ec2-user ~]$ efi-updatevar -f KEK.auth KEK
```

```
[ec2-user ~]$ efi-updatevar -f PK.auth PK
```

Untuk memverifikasi bahwa UEFI Secure Boot diaktifkan

Untuk memverifikasi bahwa UEFI Secure Boot diaktifkan, ikuti langkah-langkah di Memverifikasi apakah instans Amazon EC2 diaktifkan untuk UEFI Secure Boot.

Anda sekarang dapat mengekspor penyimpanan variabel UEFI Anda dengan perintah get-instance-uefi-data CLI, atau Anda melanjutkan ke langkah berikutnya dan menandatangani gambar boot Anda untuk reboot ke instance UEFI Secure. Boot-enabled

Langkah 3: Buat AMI dari instans

Untuk membuat AMI dari instans, Anda dapat menggunakan konsol atau API, CLI, atau `CreateImage` SDK. Untuk instruksi konsol, lihat Buat EBS-backed AMI Amazon. Untuk instruksi API, lihat CreateImage.

> **Note**
>
> API `CreateImage` secara otomatis menyalin penyimpanan variabel UEFI dari instans ke AMI. Konsol menggunakan API `CreateImage`. Setelah Anda meluncurkan instans menggunakan AMI ini, instans akan memiliki penyimpanan variabel UEFI yang sama.

Tugas 2 - Opsi B: Buat gumpalan biner yang berisi penyimpanan variabel yang telah diisi sebelumnya

Setelah Anda membuat tiga pasangan kunci, Anda dapat membuat gumpalan biner yang berisi penyimpanan variabel yang telah diisi sebelumnya yang berisi kunci UEFI Secure Boot. Atau, selesaikan langkah-langkah dithe section called “Tugas 2 - Opsi A: Tambahkan kunci ke penyimpanan variabel dari dalam instance”.

> **Warning**
>
> Anda harus menandatangani gambar boot Anda sebelum mendaftarkan kunci, jika tidak, Anda tidak akan dapat melakukan boot instans Anda.

Langkah-langkah Opsi B:

- Langkah 1: Buat penyimpanan variabel baru atau perbarui yang sudah ada
- Langkah 2: Unggah gumpalan biner pada pembuatan AMI

Langkah 1: Buat penyimpanan variabel baru atau perbarui yang sudah ada

Anda dapat membuat penyimpanan variabel offline tanpa instans yang berjalan dengan menggunakan python-uefivars. Alat ini dapat membuat penyimpanan variabel baru dari kunci Anda. Skrip saat ini mendukung format EDK2, AWS format, dan representasi JSON yang lebih mudah diedit dengan perkakas tingkat yang lebih tinggi.

Untuk membuat penyimpanan variabel offline tanpa instans yang berjalan

1. Unduh alat di tautan berikut.

```
https://github.com/awslabs/python-uefivars
```

2. Buat penyimpanan variabel baru dari kunci Anda dengan menjalankan perintah berikut. Ini akan membuat blob biner yang dikodekan base64 di.bin. *your_binary_blob* Alat ini juga mendukung pembaruan gumpalan biner dengan menggunakan `-I` parameter.

```
./uefivars.py -i none -o aws -O your_binary_blob.bin -P PK.esl -K KEK.esl --db
 db.esl --dbx dbx.esl
```

Langkah 2: Unggah gumpalan biner pada pembuatan AMI

Gunakan register-image untuk meneruskan data penyimpanan variabel UEFI Anda. Untuk parameter `--uefi-data`, tentukan gumpalan biner Anda, dan untuk parameter `--boot-mode`, tentukan `uefi`.

```
aws ec2 register-image \
    --name uefi_sb_tpm_register_image_test \
    --uefi-data $(cat your_binary_blob.bin) \
    --block-device-mappings "DeviceName=/dev/sda1,Ebs=
 {SnapshotId=snap-0123456789example,DeleteOnTermination=true}" \
    --architecture x86_64 \
    --root-device-name /dev/sda1 \
    --virtualization-type hvm \
    --ena-support \
    --boot-mode uefi
```

## Buat AWS blob biner untuk UEFI Secure Boot

Anda dapat menggunakan langkah-langkah berikut untuk mengkustomisasi variabel UEFI Secure Boot selama pembuatan AMI. KEK yang digunakan dalam langkah-langkah ini berlaku per September 2021. Jika Microsoft memperbarui KEK, Anda harus menggunakan KEK terbaru.

Untuk membuat AWS gumpalan biner

1. Buat daftar tanda tangan PK kosong.

```
touch empty_key.crt
cert-to-efi-sig-list empty_key.crt PK.esl
```

2. Unduh sertifikat KEK.

```
https://go.microsoft.com/fwlink/?LinkId=321185
```

3. Bungkus sertifikat KEK dalam daftar tanda tangan UEFI (`siglist`).

```
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output
 MS_Win_KEK.esl MicCorKEKCA2011_2011-06-24.crt
```

4. Unduh sertifikat db Microsoft.

```
https://www.microsoft.com/pkiops/certs/MicWinProPCA2011_2011-10-19.crt
https://www.microsoft.com/pkiops/certs/MicCorUEFCA2011_2011-06-27.crt
```

5. Hasilkan daftar tanda tangan db.

```
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output
 MS_Win_db.esl MicWinProPCA2011_2011-10-19.crt
sbsiglist --owner 77fa9abd-0359-4d32-bd60-28f4e78f784b --type x509 --output
 MS_UEFI_db.esl MicCorUEFCA2011_2011-06-27.crt
cat MS_Win_db.esl MS_UEFI_db.esl > MS_db.esl
```

6. Forum Antarmuka Firmware Extensible Terpadu tidak lagi menyediakan file DBX. Mereka sekarang disediakan oleh Microsoft di GitHub. Unduh pembaruan DBX terbaru dari repositori pembaruan Microsoft Secure Boot di https://github.com/microsoft/secureboot_objects.
7. Buka kemasan update-binary yang ditandatangani.

   Buat `SplitDbxContent.ps1` dengan konten skrip di bawah ini. Atau, Anda dapat menginstal skrip dari PowerShell Galeri menggunakan`Install-Script -Name SplitDbxContent`.

```
<#PSScriptInfo

.VERSION 1.0

.GUID ec45a3fc-5e87-4d90-b55e-bdea083f732d

.AUTHOR Microsoft Secure Boot Team

.COMPANYNAME Microsoft

.COPYRIGHT Microsoft

.TAGS Windows Security

.LICENSEURI
```

```
.PROJECTURI

.ICONURI

.EXTERNALMODULEDEPENDENCIES

.REQUIREDSCRIPTS

.EXTERNALSCRIPTDEPENDENCIES

.RELEASENOTES
Version 1.0: Original published version.

#>

<#
.DESCRIPTION
 Splits a DBX update package into the new DBX variable contents and the signature
 authorizing the change.
 To apply an update using the output files of this script, try:
 Set-SecureBootUefi -Name dbx -ContentFilePath .\content.bin -SignedFilePath .
\signature.p7 -Time 2010-03-06T19:17:21Z -AppendWrite'
.EXAMPLE
.\SplitDbxAuthInfo.ps1 DbxUpdate_x64.bin
#>


# Get file from script input
$file  = Get-Content -Encoding Byte $args[0]

# Identify file signature
$chop = $file[40..($file.Length - 1)]
if (($chop[0] -ne 0x30) -or ($chop[1] -ne 0x82 )) {
    Write-Error "Cannot find signature"
    exit 1
}

# Signature is known to be ASN size plus header of 4 bytes
$sig_length = ($chop[2] * 256) + $chop[3] + 4
$sig = $chop[0..($sig_length - 1)]

if ($sig_length -gt ($file.Length + 40)) {
    Write-Error "Signature longer than file size!"
    exit 1
```

```
}

# Content is everything else
$content = $file[0..39] + $chop[$sig_length..($chop.Length - 1)]

# Write signature and content to files
Set-Content -Encoding Byte signature.p7 $sig
Set-Content -Encoding Byte content.bin $content
```

Gunakan skrip untuk membongkar file DBX yang ditandatangani.

```
PS C:\Windows\system32> SplitDbxContent.ps1 .\dbx.bin
```

Ini menghasilkan dua file - `signature.p7` dan`content.bin`. Gunakan `content.bin` pada langkah berikutnya.

8. Buat penyimpanan variabel UEFI menggunakan skrip `uefivars.py`.

```
./uefivars.py -i none -o aws -O uefiblob-microsoft-keys-empty-pk.bin -P ~/PK.esl -K
 ~/MS_Win_KEK.esl --db ~/MS_db.esl  --dbx ~/content.bin
```

9. Periksa gumpalan biner dan penyimpanan variabel UEFI.

```
./uefivars.py -i aws -I uefiblob-microsoft-keys-empty-pk.bin -o json | less
```

10. Anda dapat memperbarui gumpalan dengan meneruskannya ke alat yang sama lagi.

```
./uefivars.py -i aws -I uefiblob-microsoft-keys-empty-pk.bin -o aws -O uefiblob-
microsoft-keys-empty-pk.bin -P ~/PK.esl -K ~/MS_Win_KEK.esl --db ~/MS_db.esl  --dbx
 ~/content.bin
```

Keluaran yang diharapkan

```
Replacing PK
Replacing KEK
Replacing db
Replacing dbx
```

# Gunakan enkripsi dengan EBS-backed AMI

AMI yang didukung oleh snapshot Amazon EBS dapat memanfaatkan enkripsi Amazon EBS. Snapshot data dan volume root dapat dienkripsi dan dilampirkan ke AMI. Anda dapat meluncurkan instans dan menyalin gambar dengan disertai dukungan enkripsi EBS lengkap. Parameter enkripsi untuk operasi ini didukung di semua Wilayah AWS KMS yang tersedia.

Instans EC2 dengan volume EBS terenkripsi diluncurkan dari AMI dengan cara yang sama seperti instans lainnya. Selain itu, saat Anda meluncurkan instans dari AMI yang didukung oleh snapshot EBS yang tidak dienkripsi, Anda dapat mengenkripsi beberapa atau semua volume saat peluncuran.

Seperti volume EBS, snapshot di AMI dapat dienkripsi dengan default Anda AWS KMS key, atau ke kunci yang dikelola pelanggan yang Anda tentukan. Anda harus selalu memiliki izin untuk menggunakan kunci KMS yang dipilih.

AMI dengan snapshot terenkripsi dapat dibagikan di seluruh AWS akun. Untuk informasi selengkapnya, lihat Memahami penggunaan AMI bersama di Amazon EC2.

Enkripsi dengan EBS-backed topik AMI

- Instance-launching skenario
- Image-copying skenario

## Instance-launching skenario

Instans Amazon EC2 diluncurkan dari AMI menggunakan `RunInstances` tindakan dengan parameter yang disediakan melalui pemetaan perangkat blok, baik melalui Konsol Manajemen AWS atau langsung menggunakan API atau CLI Amazon EC2. Untuk informasi selengkapnya, lihat Memblokir pemetaan perangkat untuk volume pada instans Amazon EC2. Untuk contoh pengendalian pemetaan perangkat blok dari AWS CLI, lihat Mel uncurkan, Daftar, dan Akhiri Inst ans EC2.

Secara default, tanpa parameter enkripsi yang eksplisit, tindakan `RunInstances` mempertahankan status enkripsi snapshot sumber AMI sambil memulihkan volume EBS darinya. Jika enkripsi secara default diaktifkan, semua volume yang dibuat dari AMI (baik dari snapshot terenkripsi atau tidak terenkripsi) dienkripsi. Jika enkripsi secara default tidak diaktifkan, instance mempertahankan status enkripsi AMI.

Anda juga dapat meluncurkan instans dan sekaligus menerapkan status enkripsi baru ke volume yang dihasilkan dengan menyediakan parameter enkripsi. Sebagai hasil, perilaku berikut akan muncul:

Meluncurkan tanpa parameter enkripsi

- Snapshot yang tidak terenkripsi dipulihkan ke volume yang tidak dienkripsi, kecuali jika enkripsi secara default diaktifkan, dalam hal ini semua volume yang baru dibuat akan dienkripsi.
- Snapshot terenkripsi yang Anda miliki dipulihkan ke volume yang dienkripsi ke kunci KMS yang sama.
- Snapshot terenkripsi yang tidak Anda miliki (misalnya, AMI dibagikan dengan Anda) dikembalikan ke volume yang dienkripsi oleh kunci KMS default AWS akun Anda.

Perilaku default ini dapat ditimpa dengan menyediakan parameter enkripsi. Parameter yang tersedia adalah `Encrypted` dan `KmsKeyId`. Menetapkan hanya Hasil parameter `Encrypted` dalam:

Perilaku peluncuran instans dengan **`Encrypted`** ditetapkan, tetapi tidak ada **`KmsKeyId`** yang ditentukan

- Snapshot yang tidak dienkripsi dipulihkan ke volume EBS yang dienkripsi oleh kunci KMS default akun AWS Anda.
- Snapshot terenkripsi yang Anda miliki dipulihkan ke volume yang dienkripsi ke kunci KMS yang sama. (Dengan kata lain, parameter `Encrypted` tidak memiliki efek.)
- Snapshot terenkripsi yang tidak Anda miliki (yaitu, AMI dibagikan dengan Anda) dikembalikan ke volume yang dienkripsi oleh kunci KMS default AWS akun Anda. (Dengan kata lain, parameter `Encrypted` tidak memiliki efek.)

Mengatur parameter `Encrypted` dan `KmsKeyId` memungkinkan Anda menentukan kunci KMS non-default untuk operasi enkripsi. Perilaku berikut menghasilkan:

Instans dengan **`Encrypted`** dan **`KmsKeyId`** ditetapkan

- Snapshot yang tidak dienkripsi dipulihkan ke volume EBS yang dienkripsi oleh kunci KMS yang ditentukan.
- Snapshot terenkripsi dipulihkan ke volume EBS yang dienkripsi bukan ke kunci KMS awal, melainkan ke kunci KMS yang ditentukan.

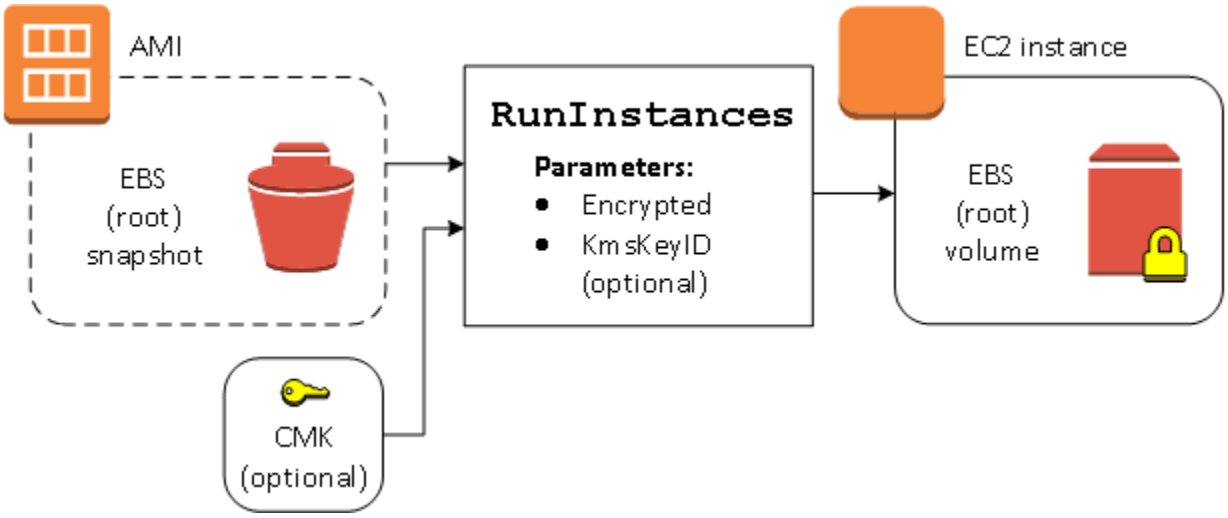
Mengirim `KmsKeyId` tanpa mengatur parameter `Encrypted` akan mengakibatkan kesalahan.

Bagian berikut ini memberikan contoh peluncuran instans dari AMI menggunakan parameter enkripsi non-default. Dalam setiap skenario ini, parameter yang diberikan ke tindakan `RunInstances` akan menghasilkan perubahan status enkripsi selama pemulihan volume dari snapshot.

Untuk informasi selengkapnya tentang meluncurkan instans dari AMI, lihat Luncurkan instans Amazon EC2.

## Mengenkripsi volume saat peluncuran

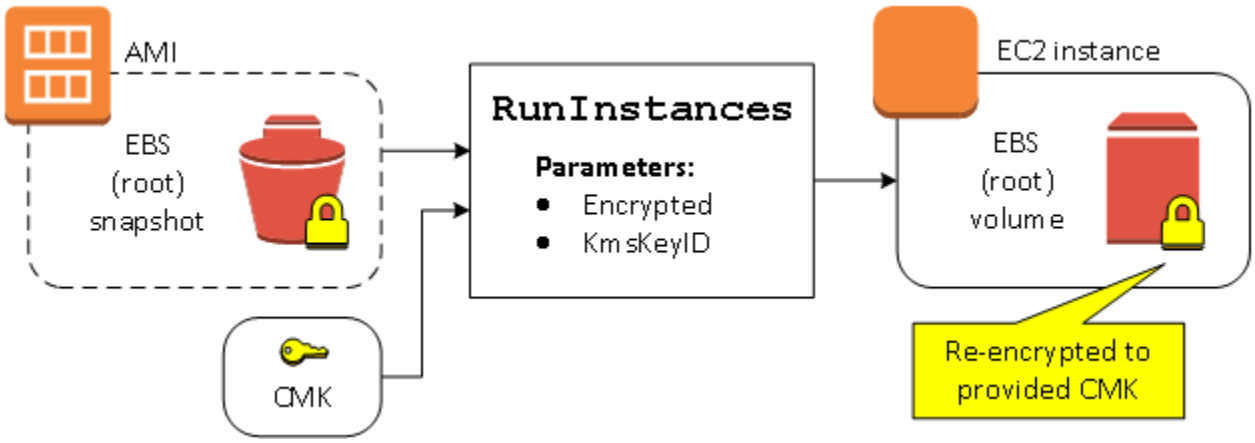
Dalam contoh ini, AMI yang didukung oleh snapshot tidak terenkripsi digunakan untuk meluncurkan instans EC2 dengan volume EBS terenkripsi.




Parameter `Encrypted` saja menyebabkan volume untuk instans ini dienkripsi. Memberikan parameter `KmsKeyId` bersifat opsional. Jika tidak ada ID kunci KMS yang ditentukan, kunci KMS default AWS akun digunakan untuk mengenkripsi volume. Untuk mengenkripsi volume ke kunci KMS berbeda yang Anda miliki, sediakan parameter `KmsKeyId`.

## Re-encrypt volume selama peluncuran

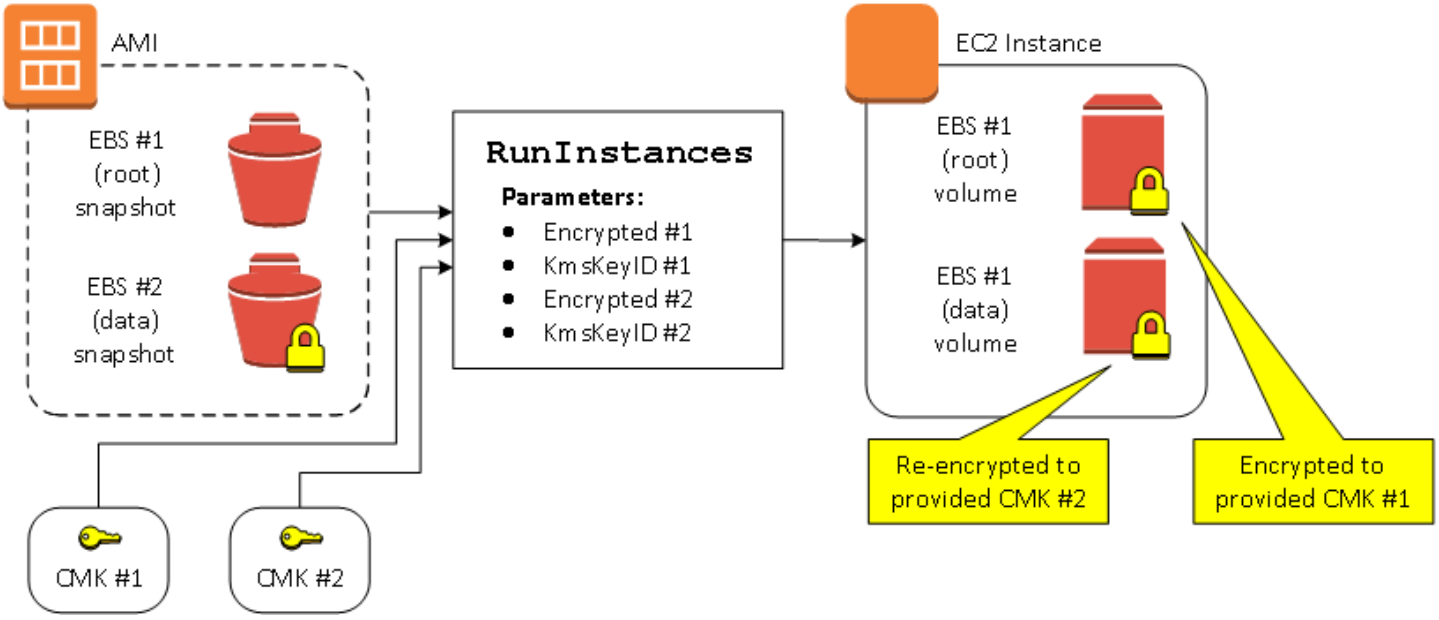
Dalam contoh ini, AMI yang didukung oleh snapshot terenkripsi digunakan untuk meluncurkan instans EC2 dengan volume EBS yang dienkripsi oleh kunci KMS baru.




Jika Anda memiliki AMI dan tidak menyediakan parameter enkripsi, instans yang dihasilkan memiliki volume yang dienkripsi oleh kunci KMS yang sama dengan snapshot. Jika AMI adalah AMI bersama, bukan milik Anda, dan Anda tidak menyediakan parameter enkripsi, volume dienkripsi oleh kunci KMS default Anda. Dengan parameter enkripsi yang disediakan seperti yang ditunjukkan, volume dienkripsi oleh kunci KMS tertentu.

## Mengubah status enkripsi beberapa volume saat peluncuran

Dalam contoh yang lebih kompleks ini, AMI yang didukung oleh beberapa snapshot (masing-masing menggunakan status enkripsi tersendiri) digunakan untuk meluncurkan instans EC2 dengan volume yang baru dienkripsi dan volume yang dienkripsi ulang.



Dalam skenario ini, tindakan `RunInstances` diberi parameter enkripsi untuk setiap snapshot sumber. Ketika semua kemungkinan parameter enkripsi ditentukan, instans yang dihasilkan adalah sama terlepas dari apakah AMI merupakan milik Anda.

## Image-copying skenario

AMI Amazon EC2 disalin menggunakan tindakan `CopyImage`, baik melalui Konsol Manajemen AWS atau secara langsung menggunakan API Amazon EC2 atau CLI.

Secara default, tanpa parameter enkripsi yang eksplisit, tindakan `CopyImage` mempertahankan status enkripsi snapshot sumber AMI selama penyalinan. Anda juga dapat menyalin AMI dan sekaligus menerapkan status enkripsi baru ke snapshot EBS terkait dengan menyediakan parameter enkripsi. Sebagai hasil, perilaku berikut akan muncul:

Menyalin tanpa parameter enkripsi

- Snapshot yang tidak terenkripsi dipulihkan ke snapshot lain yang tidak terenkripsi, kecuali jika enkripsi secara default diaktifkan, dalam hal ini semua volume yang baru dibuat akan dienkripsi.
- Snapshot terenkripsi yang Anda miliki disalin ke snapshot yang dienkripsi dengan kunci KMS yang sama.
- Snapshot terenkripsi yang tidak Anda miliki (yaitu, AMI dibagikan dengan Anda) disalin ke snapshot yang dienkripsi oleh kunci KMS default AWS akun Anda.

Perilaku default ini dapat ditimpa dengan menyediakan parameter enkripsi. Parameter yang tersedia adalah `Encrypted` dan `KmsKeyId`. Jika hanya menetapkan parameter `Encrypted`, hal berikut terjadi:

Copy-image perilaku dengan **`Encrypted`** set, tetapi tidak **`KmsKeyId`** ditentukan

- Snapshot yang tidak dienkripsi disalin ke snapshot yang dienkripsi oleh kunci KMS default akun AWS .
- Snapshot terenkripsi disalin ke snapshot yang dienkripsi oleh kunci KMS yang sama. (Dengan kata lain, parameter `Encrypted` tidak memiliki efek.)
- Snapshot terenkripsi yang tidak Anda miliki (yaitu, AMI dibagikan dengan Anda) disalin ke volume yang dienkripsi oleh kunci KMS default AWS akun Anda. (Dengan kata lain, parameter `Encrypted` tidak memiliki efek.)

Mengatur parameter `Encrypted` dan `KmsKeyId` memungkinkan Anda menentukan kunci KMS yang dikelola pelanggan untuk operasi enkripsi. Perilaku berikut menghasilkan:

Copy-image perilaku dengan keduanya **`Encrypted`** dan **`KmsKeyId`** set

- Snapshot yang tidak dienkripsi disalin ke snapshot yang dienkripsi oleh kunci KMS yang ditentukan.
- Snapshot terenkripsi disalin ke snapshot terenkripsi bukan ke kunci KMS awal, melainkan ke kunci KMS yang ditentukan.

Mengirim `KmsKeyId` tanpa turut mengatur parameter `Encrypted` akan mengakibatkan kesalahan.

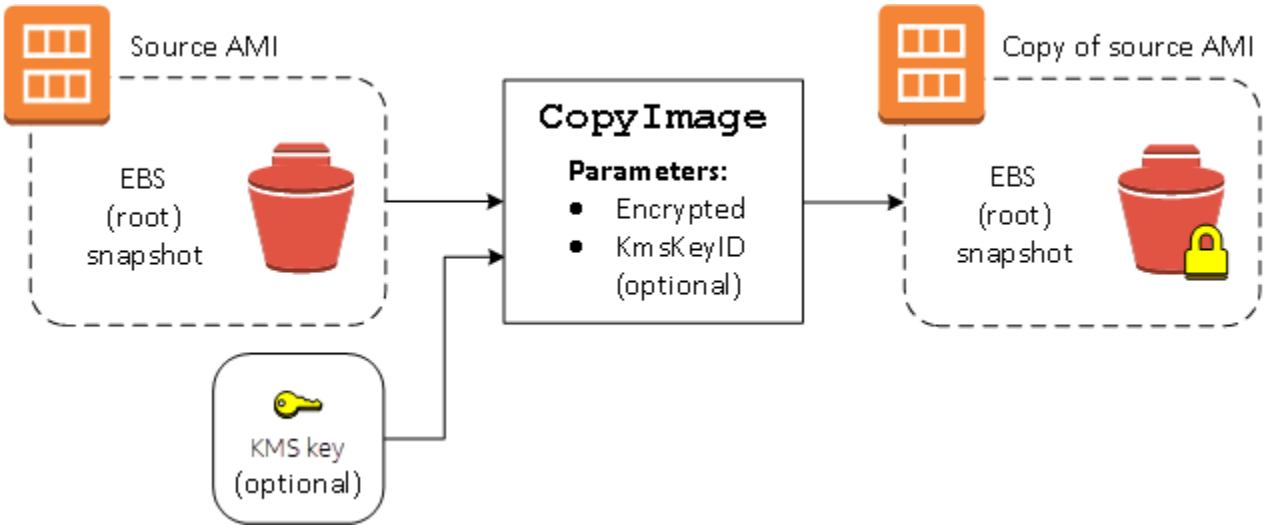
Bagian berikut ini memberikan contoh penyalinan AMI menggunakan parameter enkripsi non-default, yang menghasilkan perubahan status enkripsi.

Untuk petunjuk detail menggunakan konsol, lihat Salin AMI Amazon EC2.

## Mengenkripsikan gambar yang tidak dienkripsi selama penyalinan

Dalam skenario ini, AMI yang didukung oleh snapshot root yang tidak dienkripsi disalin ke AMI dengan snapshot root yang dienkripsi. Tindakan `CopyImage` diinvokasi dengan dua parameter enkripsi, termasuk kunci yang dikelola konsumen. Hasilnya, status enkripsi root snapshot berubah sehingga AMI target didukung oleh snapshot root yang berisi data yang sama dengan snapshot sumber, tetapi dienkripsi menggunakan kunci yang ditentukan. Anda mengeluarkan biaya penyimpanan untuk snapshot di kedua AMI, serta biaya untuk setiap instans yang Anda luncurkan dari AMI mana pun.

> **Note**
>
> Mengaktifkan enkripsi secara default memiliki efek yang sama seperti meny `Encrypted` etel parameter `true` untuk semua snapshot di AMI.




Mengatur parameter `Encrypted` akan mengenkripsi snapshot tunggal untuk instans ini. Jika Anda tidak menentukan parameter `KmsKeyId`, kunci default yang dikelola konsumen akan digunakan untuk mengenkripsi salinan snapshot.

> **Note**
>
> Anda juga dapat menyalin gambar dengan beberapa snapshot dan mengonfigurasi status enkripsi masing-masing gambar secara terpisah.

## Memahami penggunaan AMI bersama di Amazon EC2

AMI bersama adalah AMI yang dibuat oleh developer dan disediakan untuk digunakan oleh orang lain. Salah satu cara termudah untuk memulai Amazon EC2 adalah menggunakan AMI bersama yang memiliki komponen yang Anda butuhkan, lalu menambahkan konten kustom. Anda juga dapat membuat AMI Anda sendiri dan membagikannya kepada yang lain.

Anda menggunakan AMI bersama dengan risiko Anda sendiri. Amazon tidak dapat menjamin integritas atau keamanan AMI yang dibagikan oleh pengguna Amazon EC2 lainnya. Oleh karena itu, Anda harus memperlakukan AMI bersama sebagaimana halnya kode asing yang mungkin Anda pertimbangkan untuk deployment di pusat data Anda sendiri dan melakukan uji tuntas yang sesuai. Kami menyarankan Anda mendapatkan AMI dari sumber tepercaya, seperti penyedia yang terverifikasi.

# Penyedia AMI terverifikasi

Di konsol Amazon EC2, AMI publik yang dimiliki oleh Amazon atau mitra Amazon terverifikasi ditandai sebagai Penyedia terverifikasi.

Anda juga dapat menggunakan AWS CLI perintah describe-images untuk mengidentifikasi AMI publik yang berasal dari penyedia terverifikasi. Gambar publik yang dimiliki oleh Amazon atau mitra terverifikasi memiliki pemilik alias, yaituamazon,`aws-backup-vault`, atau`aws-marketplace`. Dalam output CLI, nilai-nilai ini muncul untuk `ImageOwnerAlias`. Pengguna lain tidak dapat membuat alias pada AMI tersebut. Hal ini memudahkan Anda mencari AMI dari Amazon atau mitra terverifikasi.

Untuk menjadi penyedia terverifikasi, Anda harus mendaftar sebagai penjual di AWS Marketplace. Setelah terdaftar, Anda dapat mendaftarkan AMI Anda di AWS Marketplace. Untuk informasi selengkapnya, lihat Mem ulai sebagai penjual dan AMI-based produk di Panduan AWS Marketplace Penjual.

Topik AMI bersama

- Temukan AMI bersama untuk digunakan untuk instans Amazon EC2
- Bersiaplah untuk menggunakan AMI bersama untuk Linux
- Kontrol penemuan dan penggunaan AMI di Amazon EC2 dengan AMI yang Diizinkan
- Jadikan AMI Anda tersedia untuk umum untuk digunakan di Amazon EC2
- Memahami akses publik blok untuk AMI
- Bagikan AMI dengan organisasi dan unit organisasi
- Bagikan AMI dengan spesifik AWS rekening
- Batalkan berbagi AMI dengan Anda Akun AWS
- Rekomendasi untuk membuat AMI Linux bersama

Jika Anda mencari informasi tentang topik lain

- Untuk informasi tentang membuat AMI, lihat the section called “Buat S3-backed AMI Amazon” atauthe section called “Membuat AMI”.
- Untuk informasi tentang membangun, mengirim, dan memelihara aplikasi Anda di AWS Marketplace, lihat Dokumentasi AWS Marketplace.

# Temukan AMI bersama untuk digunakan untuk instans Amazon EC2

Anda dapat menggunakan konsol Amazon EC2 atau baris perintah untuk menemukan AMI bersama publik atau pribadi untuk digunakan dengan instans Amazon EC2 Anda.

AMI adalah sumber daya Wilayah. Saat Anda mencari AMI bersama (publik atau privat), Anda harus mencarinya dari Wilayah di mana AMI tersebut dibagikan. Agar AMI tersedia di Wilayah yang berbeda, salin AMI ke Wilayah, lalu bagikan. Untuk informasi selengkapnya, lihat Salin AMI Amazon EC2.

Console

Konsol menyediakan bidang filter AMI. Anda juga dapat menelusuri pencarian Anda menggunakan filter yang disediakan di bidang Pencarian.

Untuk menemukan berbagi atau AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Di filter pertama, pilih salah satu opsi berikut:
   - Gambar pribadi — Daftar semua AMI yang dibagikan dengan Anda.
   - Gambar publik — Daftar semua AMI publik.
4. (Opsional) Untuk hanya menampilkan gambar publik dari Amazon, pilih bidang Pencarian dan kemudian, dari opsi menu, pilih alias Pemilik, lalu =, lalu amazon.
5. (Opsional) Tambahkan filter untuk menjangkau pencarian Anda ke AMI yang memenuhi persyaratan Anda.

Untuk menemukan AMI publik bersama dari penyedia terverifikasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Katalog AMI.
3. Pilih AMI Komunitas.
4. Di panel Per baiki hasil, pilih Penyedia terverifikasi. Label penyedia terverifikasi menunjukkan bahwa AMI berasal dari Amazon atau mitra terverifikasi.

AWS CLI

Gunakan perintah describe-images untuk membuat daftar AMI. Anda dapat mempersempit daftar tipe AMI yang menarik bagi Anda, seperti yang ditunjukkan dalam contoh berikut.

Untuk mencantumkan semua AMI publik

Perintah berikut mencantumkan semua AMI publik, termasuk AMI publik yang Anda miliki.

```
aws ec2 describe-images --executable-users all
```

Untuk membuat daftar AMI dengan izin peluncuran eksplisit

Perintah berikut ini mencantumkan AMI yang Anda miliki izin peluncurannya secara eksplisit. Daftar ini tidak mencakup AMI apa pun yang Anda miliki.

```
aws ec2 describe-images --executable-users self
```

Untuk daftar AMI yang dimiliki oleh penyedia terverifikasi

Perintah berikut mencantumkan AMI yang dimiliki oleh penyedia terverifikasi. AMI publik yang dimiliki oleh penyedia terverifikasi (baik Amazon atau mitra terverifikasi) memiliki pemilik alias, yang muncul sebagai`amazon`,`aws-backup-vault`, atau `aws-marketplace` di bidang akun. Hal ini membantu Anda mencari AMI dari penyedia terverifikasi dengan mudah. Pengguna lain tidak dapat membuat alias pada AMI tersebut.

```
aws ec2 describe-images \
    --owners amazon aws-marketplace \
    --query 'Images[*].[ImageId]' \
    --output text
```

Untuk mencantumkan AMI yang dimiliki oleh akun

Perintah berikut ini mencantumkan AMI yang dimiliki oleh Akun AWS tertentu.

```
aws ec2 describe-images --owners 123456789012
```

Untuk menelupkan AMI menggunakan filter

Untuk mengurangi jumlah AMI yang ditampilkan, gunakan filter untuk hanya mencantumkan tipe AMI yang menarik bagi Anda. Misalnya, gunakan filter berikut untuk hanya menampilkan EBS-backed AMI.

```
--filters "Name=root-device-type,Values=ebs"
```

PowerShell

Gunakan Get-EC2Image cmdlet (Tools for Windows PowerShell) untuk membuat daftar AMI. Anda dapat mempersempit daftar tipe AMI yang menarik bagi Anda, seperti yang ditunjukkan dalam contoh berikut.

Untuk mencantumkan semua AMI publik

Perintah berikut mencantumkan semua AMI publik, termasuk AMI publik yang Anda miliki.

```
Get-EC2Image -ExecutableUser all
```

Untuk membuat daftar AMI dengan izin peluncuran eksplisit

Perintah berikut ini mencantumkan AMI yang Anda miliki izin peluncurannya secara eksplisit. Daftar ini tidak mencakup AMI apa pun yang Anda miliki.

```
Get-EC2Image -ExecutableUser self
```

Untuk daftar AMI yang dimiliki oleh penyedia terverifikasi

Perintah berikut mencantumkan AMI yang dimiliki oleh penyedia terverifikasi. AMI publik yang dimiliki oleh penyedia terverifikasi (baik Amazon atau mitra terverifikasi) memiliki pemilik alias, yang muncul sebagai`amazon`,`aws-backup-vault`, atau `aws-marketplace` di bidang akun. Hal ini membantu Anda mencari AMI dari penyedia terverifikasi dengan mudah. Pengguna lain tidak dapat membuat alias pada AMI tersebut.

```
Get-EC2Image -Owner amazon aws-marketplace
```

Untuk mencantumkan AMI yang dimiliki oleh akun

Perintah berikut ini mencantumkan AMI yang dimiliki oleh Akun AWS tertentu.

```
Get-EC2Image -Owner 123456789012
```

Untuk menelupkan AMI menggunakan filter

Untuk mengurangi jumlah AMI yang ditampilkan, gunakan filter untuk hanya mencantumkan tipe AMI yang menarik bagi Anda. Misalnya, gunakan filter berikut untuk hanya menampilkan EBS-backed AMI.

```
-Filter @{Name="root-device-type"; Values="ebs"}
```

## Bersiaplah untuk menggunakan AMI bersama untuk Linux

Sebelum Anda menggunakan AMI bersama untuk Linux, lakukan langkah-langkah berikut untuk mengonfirmasi bahwa tidak ada kredentif pra-instal yang memungkinkan akses yang tidak diinginkan ke instans Anda oleh pihak ketiga dan tidak ada pencatatan jarak jauh yang telah dikonfigurasi sebelumnya yang dapat mengirimkan data sensitif ke pihak ketiga. Periksa dokumentasi untuk distribusi Linux yang digunakan oleh AMI untuk informasi tentang meningkatkan keamanan sistem.

Untuk memastikan Anda tidak kehilangan akses ke instans Anda tanpa sengaja, kami sarankan Anda memulai dua sesi SSH dan membuka sesi kedua hingga Anda menghapus kredensial yang tidak Anda kenali dan mengonfirmasi bahwa Anda masih dapat masuk ke aplikasi menggunakan SSH.

1. Mengidentifikasi dan menonaktifkan kunci SSH publik yang tidak sah. Satu-satunya kunci dalam file harus menjadi kunci yang Anda gunakan untuk meluncurkan AMI. Perintah berikut mencari file `authorized_keys`:

   ```
   [ec2-user ~]$ sudo find / -name "authorized_keys" -print -exec cat {} \;
   ```

2. Menonaktifkan autentikasi berbasis kata sandi untuk pengguna root. Buka file `sshd_config` dan ubah baris `PermitRootLogin` sebagai berikut:

   ```
   PermitRootLogin without-password
   ```

   Atau, Anda dapat menonaktifkan kemampuan untuk masuk ke instans sebagai pengguna root:

   ```
   PermitRootLogin No
   ```

   Memulai ulang layanan sshd.

3. Periksa apakah ada pengguna lain yang dapat masuk ke instans Anda. Pengguna dengan hak istimewa superuser sangat berbahaya. Hapus atau kunci kata sandi akun yang tidak dikenal.

4. Periksa port terbuka yang tidak Anda gunakan dan layanan jaringan yang mendengarkan koneksi masuk.
5. Untuk mencegah logging jarak jauh terkonfigurasi, Anda harus menghapus file konfigurasi yang ada dan memulai ulang layanan `rsyslog`. Sebagai contoh:

```
[ec2-user ~]$ sudo rm /etc/rsyslog.conf
[ec2-user ~]$ sudo service rsyslog restart
```

6. Pastikan bahwa semua pekerjaan cron sah.

Jika Anda menemukan AMI publik yang Anda rasa menimbulkan risiko keamanan, kontak tim keamanan AWS . Untuk informasi selengkapnya, lihat Pusat Keamanan AWS.

## Kontrol penemuan dan penggunaan AMI di Amazon EC2 dengan AMI yang Diizinkan

Untuk mengontrol penemuan dan penggunaan Amazon Machine Images (AMI) oleh pengguna di situs Anda Akun AWS, Anda dapat menggunakan fitur AMI yang Diizinkan. Anda menentukan kriteria yang harus dipenuhi AMI agar terlihat dan tersedia di dalam akun Anda. Ketika kriteria diaktifkan, pengguna yang meluncurkan instans hanya akan melihat dan memiliki akses ke AMI yang sesuai dengan kriteria yang ditentukan. Misalnya, Anda dapat menentukan daftar penyedia AMI tepercaya sebagai kriteria, dan hanya AMI dari penyedia ini yang akan terlihat dan tersedia untuk digunakan.

Sebelum mengaktifkan setelan AMI yang Diizinkan, Anda dapat mengaktifkan mode audit untuk melihat pratinjau AMI mana yang akan atau tidak akan terlihat dan tersedia untuk digunakan. Ini memungkinkan Anda menyempurnakan kriteria sesuai kebutuhan untuk memastikan bahwa hanya AMI yang dituju yang terlihat dan tersedia bagi pengguna di akun Anda. Selain itu, gunakan perintah describe-instance-image-metadata untuk menemukan instance yang diluncurkan dengan AMI yang tidak memenuhi kriteria yang ditentukan. Informasi ini dapat memandu keputusan Anda untuk memperbarui konfigurasi peluncuran Anda untuk menggunakan AMI yang sesuai (misalnya, menentukan AMI yang berbeda dalam template peluncuran) atau menyesuaikan kriteria Anda untuk mengizinkan AMI ini.

Anda menentukan setelan AMI yang Diizinkan di tingkat akun, baik secara langsung di akun atau dengan menggunakan kebijakan deklaratif. Pengaturan ini harus dikonfigurasi di setiap Wilayah AWS tempat Anda ingin mengontrol penggunaan AMI. Menggunakan kebijakan deklaratif memungkinkan Anda menerapkan pengaturan di beberapa Wilayah secara bersamaan, serta di beberapa akun

secara bersamaan. Saat kebijakan deklaratif digunakan, Anda tidak dapat mengubah pengaturan secara langsung di dalam akun. Topik ini menjelaskan cara mengonfigurasi pengaturan secara langsung di dalam akun. Untuk informasi tentang penggunaan kebijakan deklaratif, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

> **Note**
>
> Fitur AMI yang Diizinkan hanya mengontrol penemuan dan penggunaan AMI publik atau AMI yang dibagikan dengan akun Anda. Itu tidak membatasi AMI yang dimiliki oleh akun Anda. Terlepas dari kriteria yang Anda tetapkan, AMI yang dibuat oleh akun Anda selalu dapat ditemukan dan digunakan oleh pengguna di akun Anda.

Manfaat utama dari AMI yang Diizinkan

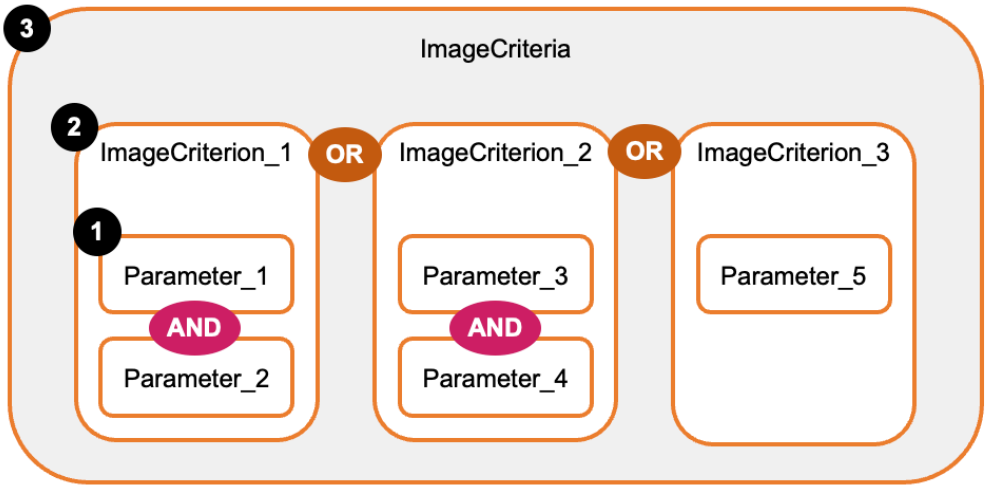
- Kepatuhan dan keamanan: Pengguna hanya dapat menemukan dan menggunakan AMI yang memenuhi kriteria yang ditentukan, sehingga mengurangi risiko penggunaan AMI yang tidak sesuai.
- Manajemen yang efisien: Dengan mengurangi jumlah AMI yang diizinkan, mengelola yang tersisa menjadi lebih mudah dan lebih efisien.
- Implementasi tingkat akun terpusat: Konfigurasikan setelan AMI yang Diizinkan di tingkat akun, baik secara langsung di dalam akun atau melalui kebijakan deklaratif. Ini menyediakan cara terpusat dan efisien untuk mengontrol penggunaan AMI di seluruh akun.

Daftar Isi


- Cara kerja AMI yang Diizinkan
- Praktik terbaik untuk menerapkan AMI yang Diizinkan
- Izin IAM yang diperlukan
- Kelola pengaturan untuk AMI yang Diizinkan


## Cara kerja AMI yang Diizinkan

Untuk mengontrol AMI mana yang dapat ditemukan dan digunakan di akun Anda, Anda menentukan serangkaian kriteria yang digunakan untuk mengevaluasi AMI. Kriteria terdiri dari satu atau lebih `ImageCriterion` seperti yang ditunjukkan pada diagram berikut. Penjelasan mengikuti diagram.




Konfigurasi memiliki tiga level:

- 1 — Nilai parameter
  - Multi-value Parameter:
    - `ImageProviders`
    - `ImageNames`
    - `MarketplaceProductCodes`

      AMI dapat mencoco kkan nilai apa pun dalam parameter yang diizinkan.

      Contoh: `ImageProviders` = `amazon` OR akun `111122223333` ATAU akun `444455556666` (Logika evaluasi untuk nilai parameter tidak ditampilkan dalam diagram.)
    - `ImageWatermarks`
  - Single-value Parameter:
    - `CreationDateCondition`
    - `DeprecationTimeCondition`
- 2 — `ImageCriterion`
  - Kelompokkan beberapa parameter dengan logika AND.
  - AMI harus cocok dengan semua parameter dalam `ImageCriterion` untuk diizinkan.
  - Contoh: `ImageProviders` = `amazon` AND `CreationDateCondition` = 300 hari atau kurang
- 3 — `ImageCriteria`
  - Kelompokkan beberapa `ImageCriterion` dengan logika OR.

- AMI dapat mencocokkan `ImageCriterion` apa pun yang diizinkan.
- Membentuk konfigurasi lengkap yang digunakan untuk mengevaluasi AMI.

Topik


- Parameter AMI yang diizinkan
- Konfigurasi AMI yang diizinkan
- Bagaimana kriteria dievaluasi
- Batas
- Operasi AMI yang diizinkan


## Parameter AMI yang diizinkan

Parameter berikut dapat dikonfigurasi untuk membuat`ImageCriterion`:

`ImageProviders`

Penyedia AMI yang AMI-nya diizinkan.

Nilai yang valid adalah alias yang ditentukan oleh AWS, dan Akun AWS ID, sebagai berikut:

- `amazon`— Alias yang mengidentifikasi AMI yang dibuat oleh Amazon atau penyedia terverifikasi
- `aws-marketplace`— Alias yang mengidentifikasi AMI yang dibuat oleh penyedia terverifikasi di AWS Marketplace
- `aws-backup-vault`— Alias yang mengidentifikasi AMI cadangan yang berada di akun brankas AWS Backup yang memiliki celah udara secara logis. Jika Anda menggunakan fitur AWS Backup secara logis air-gap vault, pastikan alias ini disertakan sebagai penyedia AMI.
- Akun AWS ID — Satu atau lebih ID 12 digit Akun AWS
- `none`— Menunjukkan bahwa hanya AMI yang dibuat oleh akun Anda yang dapat ditemukan dan digunakan. AMI publik atau bersama tidak dapat ditemukan dan digunakan. Ketika ditentukan, tidak ada kriteria lain yang dapat ditentukan.

`ImageNames`

Nama AMI yang diizinkan, menggunakan kecocokan yang tepat atau wildcard (`?`atau`*`).

`MarketplaceProductCodes`

Kode AWS Marketplace produk untuk AMI yang diizinkan.

`CreationDateCondition`

Usia maksimum untuk AMI yang diizinkan.

`DeprecationTimeCondition`

Periode maksimum sejak penghentian penggunaan untuk AMI yang diizinkan.

`ImageWatermarks`

Daftar filter watermark. Filter mengizinkan AMI jika ada tanda air pada AMI yang cocok dengan entri filter apa pun (OR logika). Setiap filter dapat menyertakan bidang berikut:

- `WatermarkKey`— Tombol tanda air untuk mencocokkan (`account-id:watermark-name`)
- `SourceImageRegion`— Wilayah AMI tempat Anda awalnya melampirkan tanda air
- `MaximumDaysSinceSourceImageCreated`— Usia maksimum sumber AMI saat Anda menerapkan tanda air
- `MaximumDaysSinceWatermarkCreated`— Jumlah hari maksimum sejak Anda menerapkan tanda air

Untuk nilai dan batasan yang valid untuk setiap kriteria, lihat ImageCriterionRequest di Refer ensi API Amazon EC2.

Konfigurasi AMI yang diizinkan

Konfigurasi inti untuk AMI yang Diizinkan adalah `ImageCriteria` konfigurasi yang menentukan kriteria untuk AMI yang diizinkan. Struktur JSON berikut menunjukkan parameter yang dapat ditentukan:

```
{
    "State": "enabled" | "disabled" | "audit-mode",
    "ImageCriteria" : [
        {
            "ImageProviders": ["string",...],
            "MarketplaceProductCodes": ["string",...],
            "ImageNames":["string",...],
            "CreationDateCondition" : {
                "MaximumDaysSinceCreated": integer
            },
            "DeprecationTimeCondition" : {
                "MaximumDaysSinceDeprecated": integer
```

```
            },
            "ImageWatermarks": [
                {
                    "WatermarkKey": "string",
                    "SourceImageRegion": "string",
                    "MaximumDaysSinceSourceImageCreated": integer,
                    "MaximumDaysSinceWatermarkCreated": integer
                },...
            ]
         },
         ...
}
```

ImageCriteria contoh

`ImageCriteria`Contoh berikut mengkonfigurasi lima`ImageCriterion`. AMI diperbolehkan jika cocok dengan salah satu dari ini`ImageCriterion`. Untuk informasi tentang bagaimana kriteria dievaluasi, lihatBagaimana kriteria dievaluasi.

```
{
    "ImageCriteria": [
        // ImageCriterion 1: Allow AWS Marketplace AMIs with product code
 "abcdefg1234567890"
        {
            "MarketplaceProductCodes": [
                "abcdefg1234567890"
            ]
        },
        // ImageCriterion 2: Allow AMIs from providers whose accounts are
        // "123456789012" OR "123456789013" AND AMI age is less than 300 days
        {
            "ImageProviders": [
                "123456789012",
                "123456789013"
            ],
            "CreationDateCondition": {
                "MaximumDaysSinceCreated": 300
            }
        },
        // ImageCriterion 3: Allow AMIs from provider whose account is "123456789014"
        // AND with names following the pattern "golden-ami-*"
        {
            "ImageProviders": [
```

```
                "123456789014"
            ],
            "ImageNames": [
                "golden-ami-*"
            ]
        },
        // ImageCriterion 4: Allow AMIs from Amazon or verified providers
        // AND which aren't deprecated
        {
            "ImageProviders": [
                "amazon"
            ],
            "DeprecationTimeCondition": {
                "MaximumDaysSinceDeprecated": 0
            }
        },
        // ImageCriterion 5: Allow AMIs that carry an approved watermark
        // from account "111122223333" AND the watermark was applied less than 90 days
 ago
        {
            "ImageWatermarks": [
                {
                    "WatermarkKey": "111122223333:prod-baseline",
                    "MaximumDaysSinceWatermarkCreated": 90
                }
            ]
        }
    ]
}
```

Bagaimana kriteria dievaluasi

Tabel berikut menjelaskan aturan evaluasi yang menentukan apakah AMI diizinkan, menunjukkan bagaimana `OR` operator `AND` or diterapkan di setiap level:

| Tingkat evaluasi | Operator | Persyaratan untuk menjadi AMI yang Diizinkan |
| --- | --- | --- |
| Nilai parameter untuk`ImageProv iders` ,`ImageName s` ,`MarketplaceProduct` | `OR` | AMI harus cocok dengan setidaknya satu nilai di setiap daftar parameter |

| Tingkat evaluasi | Operator | Persyaratan untuk menjadi AMI yang Diizinkan |
|---|---|---|
| `Codes` , dan `ImageWaterMarks` | | |
| `ImageCriterion` | `AND` | AMI harus cocok dengan semua parameter di masing-masing `ImageCriterion` |
| `ImageCriteria` | `OR` | AMI harus cocok dengan salah satu `ImageCriterion` |

Dengan menggunakan aturan evaluasi sebelumnya, mari kita lihat bagaimana menerapkannya pada[ImageCriteria contoh](#):

- `ImageCriterion1`: Mengizinkan AMI yang memiliki kode AWS Marketplace produk `abcdefg1234567890`

  `OR`
- `ImageCriterion2`: Mengizinkan AMI yang memenuhi kedua kriteria ini:
  - Dimiliki oleh salah satu akun `123456789012` `OR` `123456789013`
    - `AND`
  - Dibuat dalam 300 hari terakhir

  `OR`
- `ImageCriterion3`: Mengizinkan AMI yang memenuhi kedua kriteria ini:
  - Dimiliki oleh akun `123456789014`
    - `AND`
  - Dinamai dengan pola `golden-ami-*`

  `OR`
- `ImageCriterion4`: Mengizinkan AMI yang memenuhi kedua kriteria ini:
  - Diterbitkan oleh Amazon atau penyedia terverifikasi (ditentukan oleh `amazon` alias)
    - `AND`

- Tidak usang (hari maksimum sejak penghentian adalah) `0`

OR

- `ImageCriterion5`: Mengizinkan AMI yang membawa pencocokan tanda air:
  - Tombol tanda air `111122223333:prod-baseline`
    - AND
  - Anda menerapkan tanda air kurang dari 90 hari yang lalu

Batas

`ImageCriteria`Dapat mencakup hingga:

- 10 `ImageCriterion`

Masing-masing `ImageCriterion` dapat mencakup hingga:

- 200 nilai untuk `ImageProviders`
- 50 nilai untuk `ImageNames`
- 50 nilai untuk `MarketplaceProductCodes`
- 50 nilai untuk `ImageWatermarks`

Contoh batas

Menggunakan yang sebelumnyaImageCriteria contoh:

- Ada 5`ImageCriterion`. Hingga 5 lagi dapat ditambahkan ke permintaan untuk mencapai batas 10.
- Pada yang pertama`ImageCriterion`, ada 1 nilai untuk`MarketplaceProductCodes`. Hingga 49 lebih dapat ditambahkan `ImageCriterion` ke ini untuk mencapai batas 50.
- Yang kedua`ImageCriterion`, ada 2 nilai untuk`ImageProviders`. Hingga 198 lebih dapat ditambahkan `ImageCriterion` ke ini untuk mencapai batas 200.
- Yang ketiga`ImageCriterion`, ada 1 nilai untuk`ImageNames`. Hingga 49 lebih dapat ditambahkan `ImageCriterion` ke ini untuk mencapai batas 50.

### Operasi AMI yang diizinkan

Fitur AMI yang Diizinkan memiliki tiga status operasional untuk mengelola kriteria gambar: diaktifkan, dinonaktifkan, dan mode audit. Ini memungkinkan Anda untuk mengaktifkan atau menonaktifkan kriteria gambar, atau meninjaunya sesuai kebutuhan.

#### Diaktifkan

Saat AMI yang Diizinkan diaktifkan:

- Yang `ImageCriteria` diterapkan.
- Hanya AMI yang diizinkan yang dapat ditemukan di konsol EC2 dan oleh API yang menggunakan gambar (misalnya, yang menjelaskan, menyalin, menyimpan, atau melakukan tindakan lain yang menggunakan gambar).
- Instans hanya dapat diluncurkan menggunakan AMI yang diizinkan.

#### Nonaktif

Saat AMI yang Diizinkan dinonaktifkan:

- Itu `ImageCriteria` tidak diterapkan.
- Tidak ada batasan yang ditempatkan pada kemampuan penemuan atau penggunaan AMI.

#### Mode audit

Dalam mode audit:

- Ini `ImageCriteria` diterapkan, tetapi tidak ada batasan yang ditempatkan pada kemampuan penemuan atau penggunaan AMI.
- Di konsol EC2, untuk setiap AMI, bidang gambar yang diizinkan menampilkan Ya atau Tidak untuk menunjukkan apakah AMI akan dapat ditemukan dan tersedia bagi pengguna di akun saat AMI yang Diizinkan diaktifkan.
- Di baris perintah, respons untuk `describe-image` operasi termasuk `"ImageAllowed": true` atau `"ImageAllowed": false` untuk menunjukkan apakah AMI akan dapat ditemukan dan tersedia untuk pengguna di akun ketika AMI yang Diizinkan diaktifkan.
- Di konsol EC2, Katalog AMI menampilkan Tidak diizinkan di sebelah AMI yang tidak dapat ditemukan atau tersedia bagi pengguna di akun saat AMI yang Diizinkan diaktifkan.

## Praktik terbaik untuk menerapkan AMI yang Diizinkan

Saat menerapkan AMI yang Diizinkan, pertimbangkan praktik terbaik ini untuk memastikan transisi yang lancar dan meminimalkan potensi gangguan pada AWS lingkungan Anda.

1. Aktifkan mode audit

   Mulailah dengan mengaktifkan AMI yang Diizinkan dalam mode audit. Status ini memungkinkan Anda untuk melihat AMI mana yang akan terpengaruh oleh kriteria Anda tanpa benar-benar membatasi akses, memberikan periode evaluasi bebas risiko.
2. Tetapkan kriteria AMI yang Diizinkan

   Tentukan dengan cermat penyedia AMI mana yang selaras dengan kebijakan keamanan, persyaratan kepatuhan, dan kebutuhan operasional organisasi Anda.

   > **Note**
   >
   > Saat menggunakan layanan AWS terkelola, seperti Amazon ECS, Amazon EKS, atau Instans Terkelola AWS Lambda, sebaiknya tentukan `amazon` alias untuk mengizinkan AMI yang dibuat oleh. AWS Layanan ini bergantung pada Amazon-published AMI untuk meluncurkan instans.
   > Berhati-hatilah saat menetapkan `CreationDateCondition` batasan untuk AMI apa pun. Menyetel kondisi tanggal yang terlalu ketat (misalnya, AMI harus berusia kurang dari 5 hari) dapat menyebabkan kegagalan peluncuran instans jika AMI, baik dari penyedia AWS atau penyedia lain, tidak diperbarui dalam jangka waktu yang Anda tentukan.
   > Kami merekomendasikan memasangkan `ImageNames` dengan `ImageProviders` untuk kontrol dan spesifisitas yang lebih baik. Menggunakan `ImageNames` saja mungkin tidak secara unik mengidentifikasi AMI.
3. Periksa dampak pada proses bisnis yang diharapkan

   Anda dapat menggunakan konsol atau CLI untuk mengidentifikasi instans apa pun yang diluncurkan dengan AMI yang tidak memenuhi kriteria yang ditentukan. Informasi ini dapat memandu keputusan Anda untuk memperbarui konfigurasi peluncuran Anda untuk menggunakan AMI yang sesuai (misalnya, menentukan AMI yang berbeda dalam template peluncuran) atau menyesuaikan kriteria Anda untuk mengizinkan AMI ini.

   Konsol: Gunakan AWS Config aturan ec2-instance-launched-with-allowed-ami untuk memeriksa apakah instans yang sedang berjalan atau dihentikan diluncurkan dengan AMI yang memenuhi

kriteria AMI yang Diizinkan. Aturannya adalah NON_COM PLIANT jika AMI tidak memenuhi kriteria AMI yang Diizinkan, dan COMPLIANT jika memenuhi kriteria AMI. Aturan hanya beroperasi jika setelan AMI yang Diizinkan disetel ke mode diaktifkan atau audit.

CLI: Jalankan perintah describe-instance-image-metadata dan filter respons untuk mengidentifikasi instans apa pun yang diluncurkan dengan AMI yang tidak memenuhi kriteria yang ditentukan.

Untuk instruksi konsol dan CLI, lihatMenemukan instans yang diluncurkan dari AMI yang tidak diizinkan.

4. Aktifkan AMI yang Diizinkan

   Setelah Anda mengonfirmasi bahwa kriteria tidak akan mempengaruhi proses bisnis yang diharapkan, aktifkan AMI yang Diizinkan.

5. Peluncuran instance monitor

   Terus memantau peluncuran instans dari AMI di seluruh aplikasi Anda dan AWS layanan terkelola yang Anda gunakan, seperti Amazon EMR, Amazon ECR, Amazon EKS, dan AWS Elastic Beanstalk. Periksa masalah yang tidak terduga dan buat penyesuaian yang diperlukan pada kriteria AMI yang Diizinkan.

6. Pilot AMI baru

   Untuk menguji AMI pihak ketiga yang tidak sesuai dengan setelan AMI yang Diizinkan saat ini, rekom AWS endasikan pendekatan berikut:

   - Gunakan yang terpisah Akun AWS: Buat akun tanpa akses ke sumber daya penting bisnis Anda. Pastikan bahwa setelan AMI yang Diizinkan tidak diaktifkan di akun ini, atau bahwa AMI yang ingin Anda uji diizinkan secara eksplisit, sehingga Anda dapat mengujinya.
   - Uji di Wilayah AWS daerah lain: Gunakan Wilayah tempat AMI pihak ketiga tersedia, tetapi di mana Anda belum mengaktifkan setelan AMI yang Diizinkan.

   Pendekatan ini membantu memastikan sumber daya penting bisnis Anda tetap aman saat Anda menguji AMI baru.

## Izin IAM yang diperlukan

Untuk menggunakan fitur AMI yang Diizinkan, Anda memerlukan izin IAM berikut:

- `GetAllowedImagesSettings`

- `EnableAllowedImagesSettings`
- `DisableAllowedImagesSettings`
- `ReplaceImageCriteriaInAllowedImagesSettings`

## Kelola pengaturan untuk AMI yang Diizinkan

Anda dapat mengelola pengaturan untuk AMI yang Diizinkan. Pengaturan ini adalah per Wilayah per akun.

Tugas

- Aktifkan AMI yang Diizinkan
- Tetapkan kriteria AMI yang Diizinkan
- Nonaktifkan AMI yang Diizinkan
- Dapatkan kriteria AMI yang Diizinkan
- Temukan AMI yang diizinkan
- Menemukan instans yang diluncurkan dari AMI yang tidak diizinkan

### Aktifkan AMI yang Diizinkan

Anda dapat mengaktifkan AMI yang Diizinkan dan menentukan kriteria AMI yang Diizinkan. Kami menyarankan Anda memulai dalam mode audit, yang menunjukkan kepada Anda AMI mana yang akan terpengaruh oleh kriteria tanpa benar-benar membatasi akses.

Console

Untuk mengaktifkan AMI yang Diizinkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Peng aturan, pilih AMI yang Diizinkan.
4. Pada tab AMI yang Diizinkan, pilih Kel ola.
5. Untuk pengaturan AMI yang Diizinkan, pilih Mode Audit atau Diaktifkan. Kami menyarankan Anda memulai dalam mode audit, menguji kriteria, lalu kembali ke langkah ini untuk mengaktifkan AMI yang Diizinkan.
6. (Opsional) Untuk kriteria AMI, masukkan kriteria dalam format JSON.

7. Pilih Perbarui.

AWS CLI

Untuk mengaktifkan AMI yang Diizinkan

Gunakan perintah enable- allowed-images-settings.

```
aws ec2 enable-allowed-images-settings --allowed-images-settings-state enabled
```

Untuk mengaktifkan mode audit sebagai gantinya, tentukan al `audit-mode` ih-ali`enabled`.

```
aws ec2 enable-allowed-images-settings --allowed-images-settings-state audit-mode
```

PowerShell

Untuk mengaktifkan AMI yang Diizinkan

Gunakan Enable-EC2AllowedImagesSetting cmdlet.

```
Enable-EC2AllowedImagesSetting -AllowedImagesSettingsState enabled
```

Untuk mengaktifkan mode audit sebagai gantinya, tentukan al `audit-mode` ih-ali`enabled`.

```
Enable-EC2AllowedImagesSetting -AllowedImagesSettingsState audit-mode
```

Tetapkan kriteria AMI yang Diizinkan

Setelah mengaktifkan AMI yang Diizinkan, Anda dapat mengatur atau mengganti kriteria AMI yang Diizinkan.

Untuk konfigurasi yang benar dan nilai yang valid, lihat Konfigurasi AMI yang diizinkan danParameter AMI yang diizinkan.

Console

Untuk mengatur kriteria AMI yang Diizinkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Peng aturan, pilih AMI yang Diizinkan.
4. Pada tab AMI yang Diizinkan, pilih Kel ola.
5. Untuk kriteria AMI, masukkan kriteria dalam format JSON.
6. Pilih Perbarui.

AWS CLI

Untuk mengatur kriteria AMI yang Diizinkan

Gunakan perintah replace-image-criteria-in-allowed-images-settings dan tentukan file JSON yang berisi kriteria AMI yang diizinkan.

```
aws ec2 replace-image-criteria-in-allowed-images-settings --cli-input-json
 file://file_name.json
```

PowerShell

Untuk mengatur kriteria AMI yang Diizinkan

Gunakan Set-EC2ImageCriteriaInAllowedImagesSetting cmdlet dan tentukan file JSON yang berisi kriteria AMI yang diizinkan.

```
$imageCriteria = Get-Content -Path .\file_name.json | ConvertFrom-Json
Set-EC2ImageCriteriaInAllowedImagesSetting -ImageCriterion $imageCriteria
```

Nonaktifkan AMI yang Diizinkan

Anda dapat menonaktifkan AMI yang Diizinkan sebagai berikut.

Console

Untuk menonaktifkan AMI yang Diizinkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Peng aturan, pilih AMI yang Diizinkan.
4. Pada tab AMI yang Diizinkan, pilih Kel ola.

5. Untuk pengaturan AMI yang diizinkan, pilih Dinonaktifkan.
6. Pilih Perbarui.

AWS CLI

Untuk menonaktifkan AMI yang Diizinkan

Gunakan perintah disable-allowed-images-settings.

```
aws ec2 disable-allowed-images-settings
```

PowerShell

Untuk menonaktifkan AMI yang Diizinkan

Gunakan Disable-EC2AllowedImagesSetting cmdlet.

```
Disable-EC2AllowedImagesSetting
```

Dapatkan kriteria AMI yang Diizinkan

Anda bisa mendapatkan status setelan AMI yang Diizinkan saat ini dan kriteria AMI yang Diizinkan.

Console

Untuk mendapatkan status dan kriteria AMI yang Diizinkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Pengaturan, pilih AMI yang Diizinkan.
4. Pada tab AMI yang Diizinkan, pengaturan AMI yang Diizinkan diatur ke mode Diaktifkan, Dinonaktifkan, atau Audit.
5. Jika status AMI yang Diizinkan adalah mode Diaktifkan atau Audit, kriteria AMI, menampilkan kriteria AMI dalam format JSON.

AWS CLI

Untuk mendapatkan status dan kriteria AMI yang Diizinkan

Gunakan perintah get-allowed-images-settings.

```
aws ec2 get-allowed-images-settings
```

Dalam contoh output berikut, status `audit-mode` dan kriteria gambar diatur dalam akun.

```
{
    "State": "audit-mode",
    "ImageCriteria": [
        {
            "MarketplaceProductCodes": [
                "abcdefg1234567890"
            ]
        },
        {
            "ImageProviders": [
                "123456789012",
                "123456789013"
            ],
            "CreationDateCondition": {
                "MaximumDaysSinceCreated": 300
            }
        },
        {
            "ImageProviders": [
                "123456789014"
            ],
            "ImageNames": [
                "golden-ami-*"
            ]
        },
        {
            "ImageProviders": [
                "amazon"
            ],
            "DeprecationTimeCondition": {
                "MaximumDaysSinceDeprecated": 0
            }
        }
    ],
    "ManagedBy": "account"
}
```

PowerShell

Untuk mendapatkan status dan kriteria AMI yang Diizinkan

Gunakan Get-EC2AllowedImagesSetting cmdlet.

```
Get-EC2AllowedImagesSetting | Select-Object `
    State, `
    ManagedBy, `
    @{Name='ImageProviders'; Expression={($_.ImageCriteria.ImageProviders)}}, `
    @{Name='MarketplaceProductCodes';
 Expression={($_.ImageCriteria.MarketplaceProductCodes)}}, `
    @{Name='ImageNames'; Expression={($_.ImageCriteria.ImageNames)}}, `
    @{Name='MaximumDaysSinceCreated';
 Expression={($_.ImageCriteria.CreationDateCondition.MaximumDaysSinceCreated)}}, `
    @{Name='MaximumDaysSinceDeprecated';
 Expression={($_.ImageCriteria.DeprecationTimeCondition.MaximumDaysSinceDeprecated)}}
```

Dalam contoh output berikut, status `audit-mode` dan kriteria gambar diatur dalam akun.

```
State      : audit-mode
ManagedBy  : account
ImageProviders            : {123456789012, 123456789013, 123456789014, amazon}
MarketplaceProductCodes   : {abcdefg1234567890}
ImageNames                : {golden-ami-*}
MaximumDaysSinceCreated   : 300
MaximumDaysSinceDeprecated: 0
```

Temukan AMI yang diizinkan

Anda dapat menemukan AMI yang diizinkan atau tidak diizinkan oleh kriteria AMI yang Diizinkan saat ini.

> **Note**
>
> AMI yang diizinkan harus dalam mode audit.

Console

Untuk memeriksa apakah AMI memenuhi kriteria AMI yang Diizinkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pada tab Detail (jika Anda memilih kotak centang) atau di area ringkasan (jika Anda memilih ID AMI), temukan bidang Gambar yang diizinkan.
   - Ya — AMI memenuhi kriteria AMI yang Diizinkan. AMI ini akan tersedia untuk pengguna di akun Anda setelah Anda mengaktifkan AMI yang Diizinkan.
   - Tidak — AMI tidak memenuhi kriteria AMI yang Diizinkan.
5. Di panel navigasi, pilih Katalog AMI.

   AMI bertanda Tidak diizinkan menunjukkan AMI yang tidak memenuhi kriteria AMI yang Diizinkan. AMI ini tidak akan terlihat atau tersedia bagi pengguna di akun Anda saat AMI yang Diizinkan diaktifkan.

AWS CLI

Untuk memeriksa apakah AMI memenuhi kriteria AMI yang Diizinkan

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-id ami-0abcdef1234567890 \
    --query Images[].ImageAllowed \
    --output text
```

Berikut ini adalah output contoh.

```
True
```

Untuk menemukan AMI yang memenuhi kriteria AMI yang Diizinkan

Gunakan perintah describe-images.

```
aws ec2 describe-images \
```

```
    --filters "Name=image-allowed,Values=true" \
    --max-items 10 \
    --query Images[].ImageId
```

Berikut ini adalah output contoh.

```
ami-000eaaa8be2fd162a
ami-000f82db25e50de8e
ami-000fc21eb34c7a9a6
ami-0010b876f1287d7be
ami-0010b929226fe8eba
ami-0010957836340aead
ami-00112c992a47ba871
ami-00111759e194abcc1
ami-001112565ffcafa5e
ami-0011e45aaee9fba88
```

PowerShell

Untuk memeriksa apakah AMI memenuhi kriteria AMI yang Diizinkan

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).ImageAllowed
```

Berikut ini adalah output contoh.

```
True
```

Untuk menemukan AMI yang memenuhi kriteria AMI yang Diizinkan

Gunakan Get-EC2Image cmdlet.

```
Get-EC2Image `
    -Filter @{Name="image-allows";Values="true"} `
    -MaxResult 10 | `
    Select ImageId
```

Berikut ini adalah output contoh.

```
ami-000eaaa8be2fd162a
```

```
ami-000f82db25e50de8e
ami-000fc21eb34c7a9a6
ami-0010b876f1287d7be
ami-0010b929226fe8eba
ami-0010957836340aead
ami-00112c992a47ba871
ami-00111759e194abcc1
ami-001112565ffcafa5e
ami-0011e45aaee9fba88
```

Menemukan instans yang diluncurkan dari AMI yang tidak diizinkan

Anda dapat mengidentifikasi instans yang diluncurkan menggunakan AMI yang tidak memenuhi kriteria AMI yang Diizinkan.

Console

Untuk memeriksa apakah instance diluncurkan menggunakan AMI yang tidak diizinkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pada tab Detail, di bawah Detail Instance, temukan Gambar yang diizinkan.
   - Ya — AMI memenuhi kriteria AMI yang Diizinkan.
   - Tidak — AMI tidak memenuhi kriteria AMI yang Diizinkan.

AWS CLI

Untuk menemukan instans yang diluncurkan menggunakan AMI yang tidak diizinkan

Gunakan perintah describe-instance-image- metadata dengan filter. `image-allowed`

```
aws ec2 describe-instance-image-metadata \
    --filters "Name=image-allowed,Values=false" \
    --query "InstanceImageMetadata[*].[InstanceId,ImageMetadata.ImageId]" \
    --output table
```

Berikut ini adalah output contoh.

```
-------------------------------------------------
|          DescribeInstanceImageMetadata         |
+----------------------+-------------------------+
|  i-08fd74f3f1595fdbd |  ami-09245d5773578a1d6  |
|  i-0b1bf24fd4f297ab9 |  ami-07cccf2bd80ed467f  |
|  i-026a2eb590b4f7234 |  ami-0c0ec0a3a3a4c34c0  |
|  i-006a6a4e8870c828f |  ami-0a70b9d193ae8a799  |
|  i-0781e91cfeca3179d |  ami-00c257e12d6828491  |
|  i-02b631e2a6ae7c2d9 |  ami-0bfddf4206f1fa7b9  |
+----------------------+-------------------------+
```

PowerShell

Untuk menemukan instans yang diluncurkan menggunakan AMI yang tidak diizinkan

Gunakan Get-EC2InstanceImageMetadata cmdlet.

```
Get-EC2InstanceImageMetadata `
    -Filter @{Name="image-allowed";Values="false"} | `
    Select InstanceId, @{Name='ImageId'; Expression={($_.ImageMetadata.ImageId)}}
```

Berikut ini adalah output contoh.

```
InstanceId          ImageId
----------          -------
i-08fd74f3f1595fdbd ami-09245d5773578a1d6
i-0b1bf24fd4f297ab9 ami-07cccf2bd80ed467f
i-026a2eb590b4f7234 ami-0c0ec0a3a3a4c34c0
i-006a6a4e8870c828f ami-0a70b9d193ae8a799
i-0781e91cfeca3179d ami-00c257e12d6828491
i-02b631e2a6ae7c2d9 ami-0bfddf4206f1fa7b9
```

AWS Config

Anda dapat menambahkan AWS Config aturan ec2-instance-launched-with-allowed-ami, mengonfigurasinya sesuai kebutuhan Anda, lalu menggunakannya untuk mengevaluasi instance Anda.

Untuk informasi selengkapnya, lihat Men ambahkan AWS Config aturan dan ec2-instance-launch ed-with-allowed-ami di Panduan Pengembang. AWS Config

# Jadikan AMI Anda tersedia untuk umum untuk digunakan di Amazon EC2

Anda dapat membuat AMI Anda tersedia untuk umum dengan membagikannya kepada semua Akun AWS.

Jika Anda ingin mencegah AMI Anda dibagikan ke publik, Anda dapat mengaktifkan blokir akses publik untuk AMI. Hal ini memblokir setiap upaya untuk membuat AMI publik, membantu mencegah akses tidak sah dan potensi penyalahgunaan data AMI. Perhatikan bahwa mengaktifkan blokir akses publik tidak memengaruhi AMI Anda yang sudah tersedia untuk umum; AMI tersebut tetap tersedia untuk umum. Untuk informasi selengkapnya, lihat Memahami akses publik blok untuk AMI.

Untuk mengizinkan hanya akun tertentu yang menggunakan AMI Anda untuk meluncurkan instans, lihat Bagikan AMI dengan spesifik AWS rekening.

Daftar Isi


- Pertimbangan-pertimbangan
- Bagikan AMI dengan semua AWS akun (bagikan secara publik)


## Pertimbangan-pertimbangan

Pertimbangkan hal berikut sebelum menjadikan AMI publik.

- Kep  emilikan — Untuk membuat AMI menjadi publik, Anda Akun AWS harus memiliki AMI.
- Wilayah – AMI adalah sumber daya Wilayah. Saat Anda membagikan AMI, AMI hanya tersedia di Wilayah tempat Anda membagikannya. Agar AMI tersedia di Wilayah yang berbeda, salin AMI ke Wilayah, lalu bagikan. Untuk informasi selengkapnya, lihat Salin AMI Amazon EC2.
- Blokir akses publik – Untuk berbagi AMI secara publik, blokir akses publik untuk AMI harus dinonaktifkan di setiap Wilayah tempat AMI akan dibagikan secara publik. Setelah membagikan AMI secara publik, Anda dapat mengaktifkan kembali blokir akses publik untuk AMI untuk mencegah AMI terus dibagikan secara publik.
- Beberapa AMI tidak dapat dijadikan publik - Jika AMI Anda menyertakan salah satu komponen berikut, Anda tidak dapat menjadikannya publik (tetapi Anda dapat membagikan AMI dengan Akun AWS spesifik):
  - Volume terenkripsi
  - Snapshot volume terenkripsi
  - Kode produk

- Hindari pemaparan data sensitif – Untuk menghindari pemaparan data sensitif saat Anda membagikan AMI, baca pertimbangan keamanan di Rekomendasi untuk membuat AMI Linux bersama dan ikuti tindakan yang direkomendasikan.
- Penggunaan – Saat Anda membagikan AMI, pengguna hanya dapat meluncurkan instans dari AMI tersebut. Mereka tidak dapat menghapus, berbagi, atau memodifikasinya. Namun, setelah mereka meluncurkan instans menggunakan AMI Anda, mereka dapat membuat AMI dari instans yang mereka luncurkan.
- Pengusangan otomatis – Secara default, tanggal pengusangan semua AMI publik diatur ke dua tahun dari tanggal pembuatan AMI. Anda dapat mengatur tanggal pengusangan menjadi lebih awal dari dua tahun. Untuk membatalkan tanggal penghentian, atau memindahkan penghentian ke tanggal yang lebih baru, Anda harus menjadikan AMI pribadi dengan hanya membagikannya dengan tertentu. Akun AWS
- Hapus AMI usang — Setelah AMI publik mencapai tanggal penghentiannya, jika tidak ada instans baru yang diluncurkan dari AMI selama enam bulan atau lebih, pada AWS akhirnya akan menghapus properti berbagi publik sehingga AMI usang tidak muncul di daftar AMI publik.
- Penagihan — Anda tidak ditagih saat AMI Anda digunakan oleh orang lain Akun AWS untuk meluncurkan instans. Akun yang meluncurkan instans menggunakan AMI akan dikenai biaya untuk instans yang diluncurkan.

## Bagikan AMI dengan semua AWS akun (bagikan secara publik)

Setelah Anda menjadikan AMI publik, AMI tersedia di AMI Komunitas di konsol, yang dapat Anda akses dari Katalog AMI di navigator kiri di konsol EC2 atau saat meluncurkan instans menggunakan konsol. Perhatikan bahwa perlu waktu hingga AMI muncul di AMI Komunitas setelah Anda menjadikannya publik.

Console

Untuk menjadikan AMI publik

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI Anda dari daftar, lalu pilih Tindakan, Ubah izin AMI.
4. Di bawah Ketersediaan AMI, pilih Publik.
5. Pilih Simpan perubahan.

AWS CLI

Setiap AMI memiliki `launchPermission` properti yang mengontrol yang Akun AWS, selain milik pemilik, diizinkan untuk menggunakan AMI tersebut untuk meluncurkan instance. Dengan memodifikasi `launchPermission` properti AMI, Anda dapat membuat AMI menjadi publik (yang memberikan izin peluncuran untuk semua Akun AWS), atau membagikannya hanya dengan Akun AWS yang Anda tentukan.

Anda dapat menambahkan atau menghapus ID akun dari daftar akun yang memiliki izin peluncuran untuk AMI. Untuk menjadikan AMI publik, tentukan kelompok `all`. Anda dapat menentukan izin peluncuran publik dan eksplisit.

Untuk menjadikan AMI publik

1. Gunakan perintah modify-image-attribute sebagai berikut untuk menambahkan grup `all` ke daftar `launchPermission` untuk AMI tertentu.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission "Add=[{Group=all}]"
```

2. Untuk memverifikasi izin peluncuran AMI, gunakan perintah describe-image-attribute.

```
aws ec2 describe-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

3. (Opsional) Untuk menjadikan AMI kembali privat, hapus grup `all` dari izin peluncurannya. Perhatikan bahwa pemilik AMI selalu memiliki izin peluncuran sehingga tidak terpengaruh oleh perintah ini.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission "Remove=[{Group=all}]"
```

PowerShell

Setiap AMI memiliki `launchPermission` properti yang mengontrol yang Akun AWS, selain milik pemilik, diizinkan untuk menggunakan AMI tersebut untuk meluncurkan instance. Dengan memodifikasi `launchPermission` properti AMI, Anda dapat membuat AMI menjadi publik (yang

memberikan izin peluncuran untuk semua Akun AWS), atau membagikannya hanya dengan Akun AWS yang Anda tentukan.

Anda dapat menambahkan atau menghapus ID akun dari daftar akun yang memiliki izin peluncuran untuk AMI. Untuk menjadikan AMI publik, tentukan kelompok `all`. Anda dapat menentukan izin peluncuran publik dan eksplisit.

Untuk menjadikan AMI publik

1. Gunakan perintah Edit-EC2ImageAttribute sebagai berikut untuk menambahkan grup `all` ke daftar `launchPermission` untuk AMI tertentu.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission `
    -OperationType add `
    -UserGroup all
```

2. Untuk memverifikasi izin peluncuran AMI, gunakan perintah Get-EC2ImageAttribute berikut.

```
Get-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission
```

3. (Opsional) Untuk menjadikan AMI kembali privat, hapus grup `all` dari izin peluncurannya. Perhatikan bahwa pemilik AMI selalu memiliki izin peluncuran sehingga tidak terpengaruh oleh perintah ini.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission `
    -OperationType remove `
    -UserGroup all
```

## Memahami akses publik blok untuk AMI

Untuk mencegah pembagian publik dari AMI Anda, Anda dapat mengaktifkan  blokir akses publik untuk AMI  di tingkat akun.

Ketika mengaktifkan memblokir akses publik, setiap upaya untuk menjadikan AMI publik secara otomatis diblokir. Namun, jika Anda sudah memiliki AMI publik, mereka tetap tersedia untuk umum.

Untuk berbagi AMI secara publik, Anda harus menonaktifkan blokir akses publik. Setelah selesai berbagi, sebaiknya aktifkan kembali blokir akses publik untuk mencegah pembagian publik yang tidak diinginkan untuk AMI Anda.

> **Note**
>
> Pengaturan ini dikonfigurasi di tingkat akun, baik secara langsung di akun atau dengan menggunakan kebijakan deklaratif. Itu harus dikonfigurasi di setiap Wilayah AWS tempat Anda ingin mencegah berbagi publik dari AMI Anda. Menggunakan kebijakan deklaratif memungkinkan Anda menerapkan pengaturan di beberapa Wilayah secara bersamaan, serta di beberapa akun secara bersamaan. Saat kebijakan deklaratif digunakan, Anda tidak dapat mengubah setelan secara langsung di dalam akun. Topik ini menjelaskan cara mengonfigurasi pengaturan secara langsung di dalam akun. Untuk informasi tentang penggunaan kebijakan deklaratif, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Anda dapat membatasi izin IAM untuk pengguna administrator sehingga hanya mereka yang dapat mengaktifkan atau menonaktifkan blokir akses publik untuk AMI.

Topik


- Pengaturan default
- Kelola pengaturan akses publik blok untuk AMI


## Pengaturan default

Pengaturan Blokir akses publik untuk AMI diaktifkan atau dinonaktifkan secara default tergantung pada apakah akun Anda baru atau sudah ada, dan apakah Anda memiliki AMI publik. Tabel berikut menjelaskan pengaturan default:

| AWS akun | Blokir akses publik untuk pengaturan default AMI |
| --- | --- |
| Akun baru | Aktif |

| AWS akun | Blokir akses publik untuk pengaturan default AMI |
| --- | --- |
| Akun yang ada tanpa AMI publik [1] | Aktif |
| Akun yang ada dengan satu atau lebih AMI publik | Nonaktif |

[1] Jika akun Anda memiliki satu atau lebih AMI publik pada atau setelah 15 Juli 2023, Blokir akses publik untuk AMI dinonaktifkan secara default untuk akun Anda, bahkan jika Anda kemudian menjadikan semua AMI privat.

## Kelola pengaturan akses publik blok untuk AMI

Anda dapat mengelola setelan akses publik blokir untuk AMI Anda untuk mengontrol apakah mereka dapat dibagikan secara publik. Anda dapat mengaktifkan, menonaktifkan, atau melihat status akses publik blok saat ini untuk AMI Anda menggunakan konsol Amazon EC2 atau. AWS CLI

Melihat status blokir akses publik untuk AMI

Untuk melihat apakah pembagian publik AMI Anda diblokir di akun Anda, Anda dapat melihat status pemblokiran akses publik untuk AMI. Anda harus melihat status ini di setiap Wilayah AWS tempat Anda ingin melihat apakah pembagian publik AMI Anda diblokir.

Izin yang diperlukan

Untuk mendapatkan setelan akses publik blok saat ini untuk AMI, Anda harus memiliki izin `GetImageBlockPublicAccessState` IAM.

Console

Untuk melihat status blokir akses publik untuk AMI di Wilayah tertentu

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi (di bagian atas layar), pilih Wilayah untuk melihat status blokir akses publik untuk AMI.
3. Di panel navigasi, pilih Dasbor.
4. Pada kartu Atribut Akun, di bawah Pengaturan, pilih Perlindungan dan keamanan data.

5. Di bawah Blokir akses publik untuk AMI, periksa bidang Akses publik. Nilainya adalah Berbagi publik baru diblokir atau Berbagi publik baru diizinkan.

AWS CLI

Untuk mendapatkan status akses publik blok untuk AMI

Gunakan perintah get-image-block-public-access-state. Nilainya adalah `block-new-sharing` atau `unblocked`.

Contoh: Untuk Wilayah tertentu

```
aws ec2 get-image-block-public-access-state --region us-east-1
```

Bid `ManagedBy` ang menunjukkan entitas yang mengkonfigurasi pengaturan. Dalam contoh ini, `account` menunjukkan bahwa pengaturan dikonfigurasi langsung di akun. Nilai berarti `declarative-policy` pengaturan dikonfigurasi oleh kebijakan deklaratif. Untuk informasi selengkapnya, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

```
{
    "ImageBlockPublicAccessState": "block-new-sharing",
    "ManagedBy": "account"
}
```

Contoh: Untuk semua Wilayah di akun Anda

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 get-image-block-public-access-state \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
```

```
done
```

Berikut ini adalah output contoh.

```
Region           Public Access State
--------------   ----------------------
ap-south-1       block-new-sharing
eu-north-1       unblocked
eu-west-3        block-new-sharing
...
```

PowerShell

Untuk mendapatkan status akses publik blok untuk AMI

Gunakan Get-EC2ImageBlockPublicAccessState cmdlet. Nilainya adalah `block-new-sharing` atau `unblocked`.

Contoh: Untuk Wilayah tertentu

```
Get-EC2ImageBlockPublicAccessState -Region us-east-1
```

Berikut ini adalah output contoh.

```
block-new-sharing
```

Contoh: Untuk semua Wilayah di akun Anda

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
        [PSCustomObject]@{
            Region   = $_
            PublicAccessState = (Get-EC2ImageBlockPublicAccessState -Region $_)
        }
} | `
Format-Table -AutoSize
```

Berikut ini adalah output contoh.

```
Region         PublicAccessState
------         -----------------
```

```
ap-south-1     block-new-sharing
eu-north-1     block-new-sharing
eu-west-3      block-new-sharing
...
```

### Mengaktifkan blokir akses publik untuk AMI

Untuk mencegah AMI Anda dibagikan secara publik, aktifkan blokir akses publik untuk AMI pada tingkat akun. Anda harus mengaktifkan blokir akses publik untuk AMI di setiap Wilayah AWS tempat Anda ingin mencegah pembagian publik AMI Anda. Jika Anda sudah memiliki AMI publik, mereka akan tetap tersedia untuk umum.

#### Izin yang diperlukan

Untuk mengaktifkan setelan blokir akses publik untuk AMI, Anda harus memiliki izin `EnableImageBlockPublicAccess` IAM.

#### Pertimbangan-pertimbangan

- Diperlukan waktu hingga 10 menit untuk mengkonfigurasi pengaturan ini. Selama waktu ini, jika Anda menggambarkan status akses publik, responsnya adalah`unblocked`. Ketika konfigurasi selesai, responsnya adalah`block-new-sharing`.

#### Console

Untuk mengaktifkan blokir akses publik untuk AMI di Wilayah tertentu

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi (di bagian atas layar), pilih Wilayah tempat blokir akses publik untuk AMI diaktifkan.
3. Di panel navigasi, pilih Dasbor.
4. Pada kartu Atribut Akun, di bawah Pengaturan, pilih Perlindungan dan keamanan data.
5. Di bawah Blokir akses publik untuk AMI, pilih Kelola.
6. Pilih kotak centang Blokir berbagi publik baru, lalu pilih Per barui.

#### AWS CLI

Untuk mengaktifkan akses publik blokir untuk AMI

Gunakan perintah enable- image-block-public-access.

Contoh: Untuk Wilayah tertentu

```
aws ec2 enable-image-block-public-access \
--region us-east-1 \
--image-block-public-access-state block-new-sharing
```

Berikut ini adalah output contoh.

```
{
    "ImageBlockPublicAccessState": "block-new-sharing"
}
```

Contoh: Untuk semua Wilayah di akun Anda

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 enable-image-block-public-access \
            --region $region \
            --image-block-public-access-state block-new-sharing \
            --output text)
        echo -e "$region \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region           Public Access State
--------------   ----------------------
ap-south-1       block-new-sharing
eu-north-1       block-new-sharing
eu-west-3        block-new-sharing
...
```

PowerShell

Untuk mengaktifkan akses publik blokir untuk AMI

Gunakan perintah Enable-EC2ImageBlockPublicAccess.

Contoh: Untuk Wilayah tertentu

```
Enable-EC2ImageBlockPublicAccess `
    -Region us-east-1 `
    -ImageBlockPublicAccessState block-new-sharing
```

Berikut ini adalah output contoh.

```
Value
-----
block-new-sharing
```

Contoh: Untuk semua Wilayah di akun Anda

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region            = $_
        PublicAccessState = (
        Enable-EC2ImageBlockPublicAccess `
         -Region $_ `
         -ImageBlockPublicAccessState block-new-sharing)
    }
} | `
Format-Table -AutoSize
```

Berikut ini adalah output contoh.

```
Region         PublicAccessState
------         -----------------
ap-south-1     block-new-sharing
eu-north-1     block-new-sharing
eu-west-3      block-new-sharing
...
```

### Menonaktifkan blokir akses publik untuk AMI

Untuk memungkinkan pengguna di akun Anda membagikan AMI Anda secara publik, nonaktifkan blokir akses publik di tingkat akun. Anda harus menonaktifkan akses publik blokir untuk AMI Wilayah AWS di masing-masing tempat Anda ingin mengizinkan berbagi publik dari AMI Anda.

#### Izin yang diperlukan

Untuk menonaktifkan pengaturan akses publik blokir untuk AMI, Anda harus memiliki izin `DisableImageBlockPublicAccess` IAM.

#### Pertimbangan-pertimbangan

- Diperlukan waktu hingga 10 menit untuk mengkonfigurasi pengaturan ini. Selama waktu ini, jika Anda menggambarkan status akses publik, responsnya adalah`block-new-sharing`. Ketika konfigurasi selesai, responsnya adalah`unblocked`.

#### Console

Untuk menonaktifkan blokir akses publik untuk AMI di Wilayah tertentu

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi (di bagian atas layar), pilih Wilayah tempat blokir akses publik untuk AMI ingin dinonaktifkan.
3. Di panel navigasi, pilih Dasbor.
4. Pada kartu Atribut Akun, di bawah Pengaturan, pilih Perlindungan dan keamanan data.
5. Di bawah Blokir akses publik untuk AMI, pilih Kelola.
6. Kosongkan kotak centang Blokir berbagi publik baru, lalu pilih Per barui.
7. Masukkan **`confirm`** saat diminta konfirmasi, lalu pilih Izinkan berbagi publik.

#### AWS CLI

Untuk menonaktifkan blokir akses publik untuk AMI

Gunakan perintah disable- image-block-public-access.

Contoh: Untuk Wilayah tertentu

```
aws ec2 disable-image-block-public-access --region us-east-1
```

Berikut ini adalah output contoh.

```
{
   "ImageBlockPublicAccessState": "unblocked"
}
```

Contoh: Untuk semua Wilayah di akun Anda

```
echo -e "Region   \t Public Access State" ; \
echo -e "-------------- \t ----------------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 disable-image-block-public-access \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region           Public Access State
--------------   ----------------------
ap-south-1       unblocked
eu-north-1       unblocked
eu-west-3        unblocked
...
```

PowerShell

Untuk menonaktifkan blokir akses publik untuk AMI

Gunakan Disable-EC2ImageBlockPublicAccess cmdlet.

Contoh: Untuk Wilayah tertentu

```
Disable-EC2ImageBlockPublicAccess -Region us-east-1
```

Berikut ini adalah output contoh.

```
Value
-----
unblocked
```

Contoh: Untuk semua Wilayah di akun Anda

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region            = $_
        PublicAccessState = (Disable-EC2ImageBlockPublicAccess -Region $_)
    }
} | `
Format-Table -AutoSize
```

Berikut ini adalah output contoh.

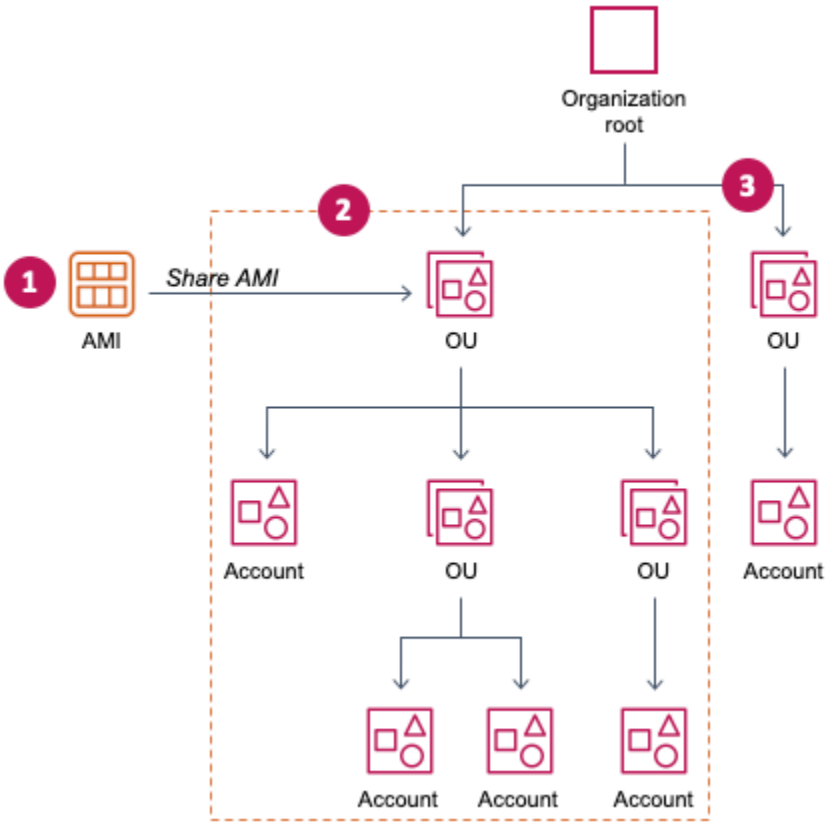
```
Region         PublicAccessState
------         -----------------
ap-south-1     unblocked
eu-north-1     unblocked
eu-west-3      unblocked
...
```

## Bagikan AMI dengan organisasi dan unit organisasi

AWS Organizationsadalah layanan manajemen akun yang memungkinkan Anda untuk mengkonsolidasikan beberapa Akun AWS ke dalam organisasi yang Anda buat dan kelola secara terpusat. Anda dapat berbagi AMI dengan organisasi atau unit organisasi (OU) yang telah Anda buat, selain membagikannya dengan akun tertentu.

Organisasi adalah entitas yang Anda buat untuk mengkonsolidasikan dan mengelola Akun AWS Anda secara terpusat. Anda dapat mengorganisasi akun dalam struktur hierarkis seperti pohon, dengan root di bagian atas dan unit-unit organisasi bersarang di bawah root organisasi. Setiap akun dapat ditambahkan langsung ke root, atau ditempatkan di salah satu OU di hierarki. Untuk informasi selengkapnya, lihat Terminologi dan konsep organisasi AWS di Panduan Pengguna AWS Organizations .

Saat Anda berbagi AMI dengan organisasi atau OU, semua akun turunan mendapatkan akses ke AMI. Misalnya, dalam diagram berikut, AMI dibagikan dengan OU tingkat atas (ditunjukkan oleh panah pada angka 1). Semua OU dan akun yang bersarang di bawah OU tingkat atas itu (ditunjukkan oleh garis putus-putus di nomor 2) juga memiliki akses ke AMI. Akun di organisasi dan OU di luar garis putus-putus (ditunjukkan oleh angka 3) tidak memiliki akses ke AMI karena mereka bukan turunan dari OU yang dibagikan AMI.




Topik

- Pertimbangan-pertimbangan
- Dapatkan ARN dari organisasi atau unit organisasi
- Mengizinkan organisasi dan OU untuk menggunakan kunci KMS
- Mengelola berbagi AMI dengan organisasi atau OU

## Pertimbangan-pertimbangan

Pertimbangkan hal-hal berikut ketika berbagi AMI dengan organisasi atau unit organisasi tertentu.

- Kepemilikan – Untuk berbagi AMI, Akun AWS Anda harus merupakan pemilik AMI.

- Batas berbagi – Pemilik AMI dapat berbagi AMI dengan organisasi atau OU mana pun, termasuk organisasi dan OU yang bukan anggotanya.

  Untuk jumlah maksimum entitas yang dapat dibagikan AMI dalam satu Wilayah, lihat Kuota layanan Amazon EC2.

- Tag – Anda tidak dapat membagikan tag buatan pengguna (tag yang Anda lampirkan ke AMI). Saat Anda berbagi AMI, tag yang ditentukan pengguna tidak tersedia untuk siapa pun Akun AWS di organisasi atau OU tempat AMI dibagikan.

- Format ARN – Saat Anda menentukan organisasi atau OU dalam sebuah perintah, pastikan menggunakan format ARN yang benar. Anda akan mendapatkan kesalahan jika Anda hanya menentukan ID, misalnya, jika Anda hanya menentukan `o-123example` atau `ou-1234-5example`.

  Format ARN yang benar:

  - ARN Organisasi: `arn:aws:organizations::`*`111122223333`*`:organization/`*`organization-id`*
  - ARN OU: `arn:aws:organizations::`*`111122223333`*`:ou/`*`organization-id`*`/`*`ou-id`*

  Di mana:

  - *`111122223333`*adalah contoh ID akun 12 digit untuk akun manajemen. Jika Anda tidak tahu nomor akun manajemen, Anda dapat menjelaskan organisasi atau unit organisasi untuk mendapatkan ARN, yang mencakup nomor akun manajemen. Untuk informasi selengkapnya, lihat Dapatkan ARN dari organisasi atau unit organisasi.
  - *`organization-id`* adalah ID organisasi, misalnya, `o-123example`.
  - *`ou-id`* adalah ID unit organisasi, misalnya, `ou-1234-5example`.

  Untuk informasi selengkapnya tentang format ARN, lihat Nama Sumber Daya Amazon (ARN) di Panduan Pengguna IAM.

- Enkripsi dan kunci – Anda dapat berbagi AMI yang didukung oleh snapshot yang tidak terenkripsi dan terenkripsi.
  - Snapshot terenkripsi harus dienkripsi dengan kunci yang dikelola pelanggan. Anda tidak dapat membagikan AMI yang didukung oleh snapshot yang dienkripsi dengan kunci terkel AWS ola default.
  - Jika Anda membagikan AMI yang didukung oleh snapshot terenkripsi, Anda harus mengizinkan organisasi atau OU untuk menggunakan kunci terkelola pelanggan yang digunakan untuk

mengenkripsi snapshot. Untuk informasi selengkapnya, lihat Mengizinkan organisasi dan OU untuk menggunakan kunci KMS.

- Wilayah – AMI adalah sumber daya Wilayah. Saat Anda membagikan AMI, AMI hanya tersedia di Wilayah tempat Anda membagikannya. Agar AMI tersedia di Wilayah yang berbeda, salin AMI ke Wilayah, lalu bagikan. Untuk informasi selengkapnya, lihat Salin AMI Amazon EC2.
- Penggunaan – Saat Anda membagikan AMI, pengguna hanya dapat meluncurkan instans dari AMI tersebut. Mereka tidak dapat menghapus, berbagi, atau memodifikasinya. Namun, setelah mereka meluncurkan instans menggunakan AMI Anda, mereka dapat membuat AMI dari instans yang mereka luncurkan.
- Penagihan — Anda tidak ditagih saat AMI Anda digunakan oleh orang lain Akun AWS untuk meluncurkan instans. Akun yang meluncurkan instans menggunakan AMI akan dikenai biaya untuk instans yang diluncurkan.

## Dapatkan ARN dari organisasi atau unit organisasi

ARN organisasi dan unit organisasi berisi nomor akun manajemen 12 digit. Jika Anda tidak tahu nomor akun manajemen, Anda dapat menjelaskan organisasi atau unit organisasi untuk mendapatkan ARN untuk keduanya. Dalam contoh berikut, `123456789012` adalah ID akun dari akun manajemen.

Izin yang diperlukan

Sebelum Anda bisa mendapatkan ARN, Anda harus memiliki izin untuk menjelaskan organisasi dan unit organisasi. Kebijakan berikut ini memberikan izin yang diperlukan.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "organizations:Describe*"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

AWS CLI

Untuk mendapatkan ARN suatu organisasi

Gunakan perintah describe-organization. Tambahkan `--query` opsi untuk mengembalikan hanya ARN organisasi.

```
aws organizations describe-organization --query 'Organization.Arn'
```

Berikut ini adalah output contoh.

```
"arn:aws:organizations::123456789012:organization/o-1234567abc"
```

Untuk mendapatkan ARN unit organisasi

Gunakan perintah describe-organizational-unit. Gunakan `--query` parameter untuk mengembalikan hanya unit organisasi ARN.

```
aws organizations describe-organizational-unit \
    --organizational-unit-id ou-a123-b4567890 \
    --query 'OrganizationalUnit.Arn'
```

Berikut ini adalah output contoh.

```
"arn:aws:organizations::123456789012:ou/o-1234567abc/ou-a123-b4567890"
```

PowerShell

Untuk mendapatkan ARN suatu organisasi

Gunakan Get-ORGOrganization cmdlet.

```
(Get-ORGOrganization).Arn
```

Berikut ini adalah output contoh.

```
arn:aws:organizations::123456789012:organization/o-1234567abc
```

Untuk mendapatkan ARN unit organisasi

Gunakan Get-ORGOrganizationalUnit cmdlet.

```
(Get-ORGOrganizationalUnit -OrganizationalUnitId "ou-a123-b4567890").Arn
```

Berikut ini adalah output contoh.

```
arn:aws:organizations::123456789012:ou/o-1234567abc/ou-a123-b4567890
```

## Mengizinkan organisasi dan OU untuk menggunakan kunci KMS

Jika Anda berbagi AMI yang didukung oleh snapshot terenkripsi, Anda juga harus mengizinkan organisasi atau unit organisasi (OU) untuk menggunakan kunci KMS yang digunakan untuk mengenkripsi snapshot.

> **Note**
>
> Snapshot terenkripsi harus dienkripsi dengan kunci yang dikelola pelanggan. Anda tidak dapat membagikan AMI yang didukung oleh snapshot yang dienkripsi dengan kunci terkel AWS ola default.

Untuk mengontrol akses ke kunci KMS, dalam kebijakan kunci Anda dapat menggunakan kunci `aws:PrincipalOrgID` dan `aws:PrincipalOrgPaths` kondisi untuk mengizinkan hanya izin prinsipal tertentu untuk tindakan yang ditentukan. Prinsipal dapat berupa pengguna, peran IAM, pengguna federasi, atau pengguna Akun AWS root.

Kunci kondisi digunakan sebagai berikut:

- `aws:PrincipalOrgID`- Mengizinkan prinsipal apa pun milik organisasi yang diwakili oleh ID yang ditentukan.
- `aws:PrincipalOrgPaths`— Mengizinkan prinsipal apa pun milik OU yang diwakili oleh jalur yang ditentukan.

Untuk memberikan izin kepada organisasi (termasuk OU dan akun miliknya) untuk menggunakan kunci KMS, tambahkan pernyataan berikut ke kebijakan kunci.

```
{
    "Sid": "Allow access for organization root",
    "Effect": "Allow",
    "Principal": "*",
    "Action": [
        "kms:Describe*",
        "kms:List*",
        "kms:Get*",
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:CreateGrant"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:PrincipalOrgID": "o-123example"
        }
    }
}
```

Untuk memberikan OU tertentu (dan akun miliknya) izin untuk menggunakan kunci KMS, Anda dapat menggunakan kebijakan yang mirip dengan contoh berikut.

```
{
        "Sid": "Allow access for specific OUs and their descendants",
        "Effect": "Allow",
        "Principal": "*",
        "Action": [
            "kms:Describe*",
            "kms:List*",
            "kms:Get*",
            "kms:Encrypt",
            "kms:Decrypt",
            "kms:ReEncrypt*",
            "kms:GenerateDataKey*",
            "kms:CreateGrant"
        ],
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "aws:PrincipalOrgID": "o-123example"
```

```
            },
            "ForAnyValue:StringLike": {
                "aws:PrincipalOrgPaths": [
                    "o-123example/r-ab12/ou-ab12-33333333/*",
                    "o-123example/r-ab12/ou-ab12-22222222/*"
                ]
            }
        }
}
```

Untuk contoh pernyataan kondisi lainnya, lihat aws:PrincipalOrgID dan aws:PrincipalOrgPaths di Panduan Pengguna IAM.

Untuk informasi tentang akses lintas akun, lihat Men gizinkan pengguna di akun lain menggunakan kunci KMS di Panduan Peng AWS Key Management Service embang.

## Mengelola berbagi AMI dengan organisasi atau OU

Anda dapat mengelola berbagi AMI dengan organisasi dan unit organisasi (OU) untuk mengontrol apakah mereka dapat meluncurkan instans Amazon EC2.

Melihat organisasi dan OU yang berbagi AMI Anda

Anda dapat menemukan organisasi dan OU yang telah berbagi AMI dengan Anda.

Console

Untuk memeriksa dengan organisasi dan OU mana yang telah Anda bagikan AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI Anda dalam daftar, pilih tab Izin, dan gulir ke bawah ke Shared organizations/OUs.

   Untuk mencari AMI yang dibagikan dengan Anda, lihat Temukan AMI bersama untuk digunakan untuk instans Amazon EC2.

AWS CLI

Untuk memeriksa dengan organisasi dan OU mana yang telah Anda bagikan AMI

Gunakan describe-image-attribute perintah dengan `launchPermission` atribut.

```
aws ec2 describe-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

Berikut ini adalah contoh respons.

```
{
    "ImageId": "ami-0abcdef1234567890",
    "LaunchPermissions": [
        {
            "OrganizationalUnitArn": "arn:aws:organizations::111122223333:ou/
o-123example/ou-1234-5example"
        }
    ]
}
```

PowerShell

Untuk memeriksa dengan organisasi dan OU mana yang telah Anda bagikan AMI

Gunakan Get-EC2ImageAttribute cmdlet.

```
Get-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission
```

### Berbagi AMI dengan organisasi atau OU

Anda dapat berbagi AMI dengan organisasi atau OU.

> **Note**
>
> Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk AMI untuk membagikan AMI. Hanya AMI itu sendiri yang perlu dibagikan, dan sistem secara otomatis menyediakan instance dengan akses ke snapshot EBS yang direferensikan untuk peluncuran. Namun, Anda perlu berbagi kunci KMS yang digunakan untuk mengenkripsi snapshot yang ditunjuk oleh AMI. Untuk informasi selengkapnya, lihat Mengizinkan organisasi dan OU untuk menggunakan kunci KMS.

Console

Untuk berbagi AMI dengan organisasi atau OU

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI Anda dalam daftar, lalu pilih Tindakan, Ubah izin AMI.
4. Di bawah Ketersediaan AMI, pilih Privat.
5. Di samping Shared organizations/OUs, pilih Add organization/OU ARN.
6. Untuk Organization/OU ARN, masukkan organisasi ARN atau OU ARN yang ingin Anda bagikan AMI, lalu pilih Bagikan AMI. Perhatikan bahwa Anda harus menentukan ARN lengkap, bukan hanya ID-nya.

   Untuk membagikan AMI ini kepada beberapa organisasi atau OU, ulangi langkah ini hingga Anda telah menambahkan semua organisasi atau OU yang diperlukan.
7. Setelah selesai, pilih Simpan perubahan.
8. (Opsional) Untuk melihat organisasi atau OU yang telah berbagi AMI dengan Anda, pilih AMI dalam daftar, pilih tab Izin, dan gulir ke bawah ke Shared organizations/OUs. Untuk mencari AMI yang dibagikan dengan Anda, lihat Temukan AMI bersama untuk digunakan untuk instans Amazon EC2.

AWS CLI

Untuk berbagi AMI dengan organisasi

Gunakan modify-image-attribute perintah untuk memberikan izin peluncuran untuk AMI yang ditentukan ke organisasi yang ditentukan.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission
 "Add=[{OrganizationArn=arn:aws:organizations::123456789012:organization/
o-123example}]"
```

Untuk berbagi AMI dengan OU

Perintah modify-image-attribute memberikan izin peluncuran untuk AMI yang ditentukan ke OU tertentu. Perhatikan bahwa Anda harus menentukan ARN lengkap, bukan hanya ID-nya.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission
 "Add=[{OrganizationalUnitArn=arn:aws:organizations::123456789012:ou/o-123example/
ou-1234-5example}]"
```

PowerShell

Gunakan Edit-EC2ImageAttribute perintah (Tools for Windows PowerShell) untuk berbagi AMI seperti yang ditunjukkan dalam contoh berikut.

Untuk berbagi AMI dengan organisasi atau OU

Perintah berikut memberikan izin peluncuran untuk AMI yang ditentukan ke organisasi tertentu.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission `
    -OperationType add `
    -OrganizationArn "arn:aws:organizations::123456789012:organization/o-123example"
```

Untuk berhenti berbagi AMI dengan organisasi atau OU

Perintah berikut menghapus izin peluncuran untuk AMI yang ditentukan dari organisasi tertentu:

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission `
    -OperationType remove `
    -OrganizationArn "arn:aws:organizations::123456789012:organization/o-123example"
```

Untuk berhenti berbagi AMI dengan semua organisasi, OU, dan Akun AWS

Perintah berikut ini menghapus semua izin peluncuran publik dan eksplisit dari AMI yang ditentukan. Perhatikan bahwa pemilik AMI selalu memiliki izin peluncuran sehingga tidak terpengaruh oleh perintah ini.

```
Reset-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission
```

### Berhenti berbagi AMI dengan organisasi atau OU

Anda dapat berhenti berbagi AMI dengan organisasi atau OU.

> **Note**
>
> Anda tidak dapat berhenti berbagi AMI dengan akun tertentu jika akun tersebut berada di organisasi atau OU yang dengannya AMI dibagikan. Jika Anda mencoba untuk berhenti berbagi AMI dengan menghapus izin peluncuran untuk akun tersebut, Amazon EC2 akan menampilkan pesan sukses. Namun, AMI terus dibagikan dengan akun tersebut.

#### Console

Untuk berhenti berbagi AMI dengan organisasi atau OU

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI Anda dalam daftar, lalu pilih Tindakan, Ubah izin AMI.
4. Di bawah organizations/OUs Bersama, pilih organisasi atau OU yang ingin Anda hentikan berbagi AMI, lalu pilih H apus yang dipilih.
5. Setelah selesai, pilih Simpan perubahan.
6. (Opsional) Untuk mengonfirmasi bahwa Anda telah berhenti berbagi AMI dengan organisasi atau OU, pilih AMI dalam daftar, pilih tab Izin, dan gulir ke bawah ke Shared organizations/OUs.

#### AWS CLI

Untuk berhenti berbagi AMI dengan organisasi atau OU

Gunakan perintah modify- image-atribut. Contoh ini menghapus izin peluncuran untuk AMI yang ditentukan dari organisasi yang ditentukan.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission
 "Remove=[{OrganizationArn=arn:aws:organizations::123456789012:organization/
o-123example}]"
```

Untuk berhenti berbagi AMI dengan semua organisasi, OU, dan Akun AWS

Gunakan perintah reset-image-attribute. Contoh ini menghapus semua izin peluncuran publik dan eksplisit dari AMI yang ditentukan. Perhatikan bahwa pemilik AMI selalu memiliki izin peluncuran sehingga tidak terpengaruh oleh perintah ini.

```
aws ec2 reset-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

PowerShell

Untuk berhenti berbagi AMI dengan organisasi atau OU

Gunakan Edit-EC2ImageAttribute cmdlet. Contoh ini menghapus izin peluncuran untuk AMI yang ditentukan dari organisasi yang ditentukan.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission `
    -OperationType remove `
    -OrganizationArn "arn:aws:organizations::123456789012:organization/o-123example"
```

Untuk berhenti berbagi AMI dengan semua organisasi, OU, dan Akun AWS

Gunakan Reset-EC2ImageAttribute cmdlet. Contoh ini menghapus semua izin peluncuran publik dan eksplisit dari AMI yang ditentukan. Perhatikan bahwa pemilik AMI selalu memiliki izin peluncuran sehingga tidak terpengaruh oleh perintah ini.

```
Reset-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute LaunchPermission
```

## Bagikan AMI dengan spesifik AWS rekening

Anda dapat berbagi AMI dengan spesifik Akun AWS tanpa membuat AMI menjadi publik. Yang Anda butuhkan hanyalah Akun AWS ID.

Akun AWS ID adalah angka 12 digit, seperti`012345678901`, yang secara unik mengidentifikasi sebuah Akun AWS. Untuk informasi selengkapnya, lihat Melihat Akun AWS pen gidentifikasi di Panduan AWS Account Management Referensi.

## Pertimbangan-pertimbangan

Pertimbangkan hal berikut saat berbagi AMI dengan spesifik Akun AWS.

- Kepemilikan – Untuk berbagi AMI, Akun AWS Anda harus merupakan pemilik AMI.
- Batas berbagi – Untuk jumlah maksimum entitas yang dapat digunakan bersama AMI dalam suatu Wilayah, lihat kuota layanan Amazon EC2.
- Tag – Anda tidak dapat membagikan tag buatan pengguna (tag yang Anda lampirkan ke AMI). Saat Anda berbagi AMI, tag yang ditentukan pengguna Anda tidak tersedia untuk siapa pun Akun AWS yang dibagikan AMI.
- Snapshot — Anda tidak perlu membagikan snapshot Amazon EBS yang dirujuk AMI untuk berbagi AMI. Anda hanya dapat berbagi AMI itu sendiri; sistem menyediakan akses instance ke snapshot EBS yang direferensikan untuk peluncuran. Namun, Anda harus membagikan kunci KMS apa pun yang digunakan untuk mengenkripsi snapshot yang referensi AMI. Untuk informasi selengkapnya, lihat Bagikan snapshot Amazon EBS di Panduan Pengguna Amazon EBS.
- Enkripsi dan kunci – Anda dapat berbagi AMI yang didukung oleh snapshot yang tidak terenkripsi dan terenkripsi.
  - Snapshot terenkripsi harus dienkripsi dengan kunci KMS. Anda tidak dapat berbagi AMI yang didukung oleh snapshot yang dienkripsi dengan kunci yang dikelola AWS secara default.
  - Jika Anda berbagi AMI yang didukung oleh snapshot terenkripsi, Anda harus mengizink Akun AWS annya menggunakan kunci KMS yang digunakan untuk mengenkripsi snapshot. Untuk informasi selengkapnya, lihat Mengizinkan organisasi dan OU untuk menggunakan kunci KMS. Untuk mengatur kebijakan kunci yang Anda perlukan untuk meluncurkan instans Penskalaan Otomatis saat Anda menggunakan kunci yang dikelola pelanggan untuk enkripsi, lihat AWS KMS key Kebijakan wajib untuk digunakan dengan volume terenkripsi di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
- Wilayah – AMI adalah sumber daya Wilayah. Saat Anda membagikan AMI, AMI hanya tersedia di Wilayah tersebut. Agar AMI tersedia di Wilayah yang berbeda, salin AMI ke Wilayah, lalu bagikan. Untuk informasi selengkapnya, lihat Salin AMI Amazon EC2.
- Penggunaan – Saat Anda membagikan AMI, pengguna hanya dapat meluncurkan instans dari AMI tersebut. Mereka tidak dapat menghapus, berbagi, atau memodifikasinya. Namun, setelah mereka

meluncurkan instans menggunakan AMI Anda, mereka akan dapat membuat AMI dari instans mereka.

- Menyalin AMI bersama – Jika pengguna di akun lain ingin menyalin AMI bersama, Anda harus memberi mereka izin baca untuk penyimpanan yang mendukung AMI. Untuk informasi selengkapnya, lihat Cross-account menyalin.
- Penagihan — Anda tidak ditagih saat AMI Anda digunakan oleh orang lain Akun AWS untuk meluncurkan instans. Akun yang meluncurkan instans menggunakan AMI akan dikenai biaya untuk instans yang diluncurkan.

Console

Untuk memberikan izin peluncuran eksplisit

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI Anda dalam daftar, lalu pilih Tindakan, Ubah izin AMI.
4. Pilih Privat.
5. Di bawah Akun bersama, pilih Tambahkan ID akun.
6. Untuk Akun AWS ID, masukkan Akun AWS ID yang ingin Anda gunakan untuk berbagi AMI, lalu pilih Bagikan AMI.

   Untuk membagikan AMI ini ke beberapa akun, ulangi Langkah 5 dan 6 hingga Anda telah menambahkan semua ID akun yang diperlukan.
7. Setelah selesai, pilih Simpan perubahan.
8. (Opsional) Untuk melihat Akun AWS ID yang telah Anda gunakan untuk berbagi AMI, pilih AMI dalam daftar, dan pilih tab Izin. Untuk mencari AMI yang dibagikan dengan Anda, lihat Temukan AMI bersama untuk digunakan untuk instans Amazon EC2.

AWS CLI

Gunakan modify-image-attribute perintah untuk berbagi AMI seperti yang ditunjukkan dalam contoh berikut.

Untuk memberikan izin peluncuran eksplisit

Contoh berikut memberikan izin peluncuran untuk AMI yang ditentukan ke yang ditentukan Akun AWS.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission "Add=[{UserId=123456789012}]"
```

Untuk menghapus izin peluncuran bagi sebuah akun

Contoh berikut menghapus izin peluncuran untuk AMI yang ditentukan dari yang ditentukan Akun AWS.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --launch-permission "Remove=[{UserId=123456789012}]"
```

Untuk menghapus semua izin peluncuran

Contoh berikut menghapus semua izin peluncuran publik dan eksplisit dari AMI yang ditentukan. Perhatikan bahwa pemilik AMI selalu memiliki izin peluncuran sehingga tidak terpengaruh oleh perintah ini.

```
aws ec2 reset-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute launchPermission
```

PowerShell

Gunakan Edit-EC2ImageAttribute perintah (Tools for Windows PowerShell) untuk berbagi AMI seperti yang ditunjukkan dalam contoh berikut.

Untuk memberikan izin peluncuran eksplisit

Contoh berikut memberikan izin peluncuran untuk AMI yang ditentukan ke yang ditentukan Akun AWS.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission `
    -OperationType add `
```

```
    -UserId "123456789012"
```

Untuk menghapus izin peluncuran bagi sebuah akun

Contoh berikut menghapus izin peluncuran untuk AMI yang ditentukan dari yang ditentukan Akun AWS.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission -OperationType remove `
    -UserId "123456789012"
```

Untuk menghapus semua izin peluncuran

Contoh berikut menghapus semua izin peluncuran publik dan eksplisit dari AMI yang ditentukan. Perhatikan bahwa pemilik AMI selalu memiliki izin peluncuran sehingga tidak terpengaruh oleh perintah ini.

```
Reset-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute launchPermission
```

## Batalkan berbagi AMI dengan Anda Akun AWS

Amazon Machine Image (AMI) dapat dibagikan dengan Akun AWS spesifik dengan menambahkan akun ke izin peluncuran AMI. Jika AMI telah dibagikan dengan Anda Akun AWS dan Anda tidak lagi ingin AMI dibagikan dengan akun Anda, Anda dapat menghapus akun Anda dari izin peluncuran AMI. Anda melakukan ini dengan menjalankan `cancel-image-launch-permission` AWS CLI perintah. Saat menjalankan perintah ini, Anda Akun AWS dihapus dari izin peluncuran untuk AMI yang ditentukan. Untuk menemukan AMI yang dibagikan dengan Anda Akun AWS, lihatTemukan AMI bersama untuk digunakan untuk instans Amazon EC2.

Anda dapat membatalkan AMI dibagikan dengan akun Anda, misalnya, untuk mengurangi kemungkinan peluncuran instans dengan AMI yang tidak digunakan atau sudah usang yang dibagikan kepada Anda. Saat Anda membatalkan AMI yang dibagikan dengan akun Anda, AMI tidak lagi muncul di daftar AMI apa pun di konsol EC2 atau output describe-images.

Topik

- Batasan

- Membatalkan berbagi AMI dengan akun Anda

## Batasan

- Anda dapat menghapus akun Anda dari izin peluncuran AMI yang dibagikan dengan Akun AWS satu-satunya. Anda tidak dapat menggunakannya `cancel-image-launch-permission` untuk menghapus akun Anda dari izin peluncuran AMI yang dibagikan dengan organisasi atau unit organisasi (OU) atau untuk menghapus akses ke AMI publik.
- Anda tidak dapat menghapus akun secara permanen dari izin peluncuran AMI. Pemilik AMI dapat membagikan AMI kepada akun Anda lagi.
- AMI adalah sumber daya Wilayah. Saat menjalankan `cancel-image-launch-permission`, Anda harus menentukan Wilayah tempat AMI berada. Entah menentukan Wilayah dalam perintah, atau gunakan variabel AWS_DEFAULT_REGION lingkungan.
- Hanya SDK AWS CLI dan yang mendukung penghapusan akun Anda dari izin peluncuran AMI. Konsol EC2 sekarang tidak mendukung tindakan ini.

## Membatalkan berbagi AMI dengan akun Anda

> **Note**
>
> Setelah membatalkan AMI yang dibagikan dengan akun Anda, Anda tidak dapat mengembalikannya. Untuk mendapatkan kembali akses ke AMI, pemilik AMI harus membagikannya dengan akun Anda.

### AWS CLI

Untuk membatalkan berbagi AMI dengan akun Anda

Gunakan perintah cancel-image-launch-permission.

```
aws ec2 cancel-image-launch-permission \
    --image-id ami-0abcdef1234567890 \
    --region us-east-1
```

### PowerShell

Untuk membatalkan berbagi AMI dengan akun Anda

Gunakan Stop-EC2ImageLaunchPermission cmdlet.

```
Stop-EC2ImageLaunchPermission `
    -ImageId ami-0abcdef1234567890 `
    -Region us-east-1
```

## Rekomendasi untuk membuat AMI Linux bersama

Gunakan panduan berikut ini untuk mengurangi permukaan serangan dan meningkatkan keandalan AMI yang Anda buat.

> **Important**
>
> Tidak ada daftar pedoman keamanan yang lengkap. Bangun AMI bersama Anda dengan cermat dan luangkan waktu untuk mempertimbangkan di mana Anda data sensitif Anda terekspos.

Daftar Isi


- Nonaktifkan login jarak jauh berbasis kata sandi untuk pengguna root
- Nonaktifkan akses root lokal
- Hapus pasangan kunci host SSH
- Instal kredensial kunci publik
- Nonaktifkan pemeriksaan DNS sshd (opsional)
- Hapus data sensitif


Jika Anda membuat AMI untuk AWS Marketplace, lihat Prakti k terbaik untuk membuat AMI di Panduan AWS Marketplace Penjual untuk panduan, kebijakan, dan praktik terbaik.

### Nonaktifkan login jarak jauh berbasis kata sandi untuk pengguna root

Menggunakan kata sandi root tetap untuk AMI publik adalah risiko keamanan yang segera diketahui. Bahkan mengandalkan pengguna untuk mengubah kata sandi setelah masuk pertama membuka jendela kesempatan kecil potensi penyalahgunaan.

Untuk mengatasi masalah ini, nonaktifkan masuk jarak jauh berbasis kata sandi untuk pengguna root.

Untuk menonaktifkan login jarak jauh berbasis kata sandi untuk pengguna root

1. Buka file `/etc/ssh/sshd_config` dengan teks editor dan temukan baris berikut:

   ```
   #PermitRootLogin yes
   ```

2. Ubah baris menjadi:

   ```
   PermitRootLogin without-password
   ```

   Lokasi file konfigurasi ini mungkin berbeda untuk distribusi Anda, atau jika Anda tidak menjalankan OpenSSH. Jika demikian, berkonsultasilah dengan dokumentasi yang relevan.

## Nonaktifkan akses root lokal

Saat Anda bekerja dengan AMI bersama, praktik terbaiknya adalah menonaktifkan masuk lewat root langsung. Caranya, masuk ke instans yang sedang berjalan dan keluarkan perintah berikut:

```
[ec2-user ~]$ sudo passwd -l root
```

> **Note**
>
> Perintah ini tidak memengaruhi penggunaan `sudo`.

## Hapus pasangan kunci host SSH

Jika Anda berencana untuk berbagi AMI yang berasal dari AMI publik, hapus pasangan kunci host SSH yang ada terletak di `/etc/ssh`. Hal ini memaksa SSH untuk menghasilkan pasangan kunci SSH yang baru dan unik ketika seseorang meluncurkan suatu instans menggunakan AMI Anda, yang meningkatkan keamanan dan mengurangi kemungkinan serangan "perantara".

Hapus semua file kunci berikut yang ada di sistem Anda.

- ssh_host_dsa_key
- ssh_host_dsa_key.pub
- ssh_host_key
- ssh_host_key.pub

- ssh_host_rsa_key
- ssh_host_rsa_key.pub
- ssh_host_ecdsa_key
- ssh_host_ecdsa_key.pub
- ssh_host_ed25519_key
- ssh_host_ed25519_key.pub

Anda dapat menghapus semua file ini dengan aman dengan perintah berikut.

```
[ec2-user ~]$ sudo shred -u /etc/ssh/*_key /etc/ssh/*_key.pub
```

**Warning**

Utilitas penghapusan aman seperti shred mungkin tidak menghapus semua salinan file dari media penyimpanan Anda. Salinan file tersembunyi dapat dibuat dengan penjurnalan sistem file (termasuk Amazon Linux default ext4), snapshot, pencadangan, RAID, dan cache sementara. Untuk informasi selengkapnya, lihat dokumen asi shred.

**Important**

Jika Anda lupa menghapus pasangan kunci host SSH yang ada dari AMI publik, proses audit rutin kami memberi tahu Anda dan semua pelanggan yang menjalankan instans AMI Anda tentang potensi risiko keamanan. Setelah masa tenggang singkat, kami menandai AMI sebagai privat.

## Instal kredensial kunci publik

Setelah mengonfigurasi AMI untuk mencegah masuk menggunakan kata sandi, Anda harus memastikan pengguna dapat masuk menggunakan mekanisme lain.

Amazon EC2 memungkinkan pengguna untuk menentukan nama pasangan kunci publik-privat saat meluncurkan sebuah instans. Saat nama pasangan kunci yang valid diberikan ke panggilan API `RunInstances` (atau melalui alat API baris perintah), kunci publik (bagian dari pasangan kunci yang

dimiliki Amazon EC2 di server setelah panggilan `CreateKeyPair` atau `ImportKeyPair`) tersedia untuk instans melalui kueri HTTP terhadap metadata instans.

Untuk masuk melalui SSH, AMI Anda harus mengambil nilai kunci saat boot dan menambahkannya ke `/root/.ssh/authorized_keys` (atau setara untuk akun pengguna lain di AMI). Pengguna dapat meluncurkan instans AMI Anda dengan pasangan kunci dan masuk tanpa yang memerlukan kata sandi root.

Banyak distribusi, termasuk Amazon Linux dan Ubuntu, menggunakan paket `cloud-init` untuk menginjeksikan kredensial kunci publik untuk pengguna yang telah dikonfigurasi. Jika distribusi Anda tidak mendukung `cloud-init`, Anda dapat menambahkan kode berikut ke skrip penyalaan sistem (seperti `/etc/rc.local`) untuk menarik kunci publik yang Anda tentukan pada saat peluncuran untuk pengguna root.

> **Note**
>
> Dalam contoh berikut, alamat IP http://169.254.169.254/ adalah alamat link-local dan hanya valid dari instance.

IMDSv2

```
if [ ! -d /root/.ssh ] ; then
        mkdir -p /root/.ssh
        chmod 700 /root/.ssh
fi
# Fetch public key using HTTP
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-
token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/openssh-key > /tmp/my-key
if [ $? -eq 0 ] ; then
        cat /tmp/my-key >> /root/.ssh/authorized_keys
        chmod 700 /root/.ssh/authorized_keys
        rm /tmp/my-key
fi
```

IMDSv1

```
if [ ! -d /root/.ssh ] ; then
```

```
        mkdir -p /root/.ssh
        chmod 700 /root/.ssh
fi
# Fetch public key using HTTP
curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key > /tmp/my-key
if [ $? -eq 0 ] ; then
        cat /tmp/my-key >> /root/.ssh/authorized_keys
        chmod 700 /root/.ssh/authorized_keys
        rm /tmp/my-key
fi
```

Hal ini dapat diterapkan ke pengguna; Anda tidak perlu membatasinya ke pengguna `root`.

> **Note**
>
> Pemaketan ulang instans berdasarkan AMI ini mencakup kunci yang digunakan untuk meluncurkan. Untuk mencegah inklusi kunci, Anda harus membersihkan (atau menghapus) file `authorized_keys` atau mengecualikan file ini dari pemaketan ulang.

## Nonaktifkan pemeriksaan DNS sshd (opsional)

Menonaktifkan pemeriksaan DNS sshd akan sedikit melemahkan keamanan sshd. Namun, jika resolusi DNS gagal, login SSH masih berfungsi. Jika Anda tidak menonaktifkan pemeriksaan sshd, kegagalan resolusi DNS akan mencegah semua login.

Untuk menonaktifkan pemeriksaan DNS sshd

1. Buka file `/etc/ssh/sshd_config` dengan editor teks dan cari baris berikut:

   ```
   #UseDNS yes
   ```

2. Ubah baris menjadi:

   ```
   UseDNS no
   ```

> **Note**
>
> Lokasi file konfigurasi ini mungkin berbeda untuk distribusi Anda, atau jika Anda tidak menjalankan OpenSSH. Jika demikian, berkonsultasilah dengan dokumentasi yang relevan.

## Hapus data sensitif

Kami tidak sarankan Anda menyimpan data atau perangkat lunak sensitif di AMI apa pun yang Anda bagikan. Pengguna yang meluncurkan AMI bersama mungkin dapat memaketkannya kembali dan mendaftarkannya sebagai milik mereka sendiri. Ikuti panduan ini untuk membantu Anda menghindari beberapa risiko keamanan yang mudah diabaikan:

- Kami menyarankan penggunaan opsi `--exclude` *`directory`* pada `ec2-bundle-vol` untuk melewati direktori dan subdirektori yang berisi informasi rahasia yang tidak ingin Anda sertakan dalam paketan Anda. Secara khusus, kecualikan semua pasangan public/private kunci SSH milik pengguna dan `authorized_keys` file SSH saat menggabungkan gambar. AMI publik Amazon menyimpannya di `/root/.ssh` untuk pengguna root, dan `/home/`*`user_name`*`/.ssh/` untuk pengguna biasa. Untuk informasi selengkapnya, lihat ec2-bundle-vol.
- Selalu hapus riwayat shell sebelum pembuatan paketan. Jika Anda mencoba lebih dari satu unggahan paketan di AMI yang sama, riwayat shell berisi kunci akses Anda. Contoh berikut harus menjadi perintah terakhir yang Anda jalankan sebelum memaketkan dari dalam instans.

```
[ec2-user ~]$ shred -u ~/.*history
```

> **Warning**
>
> Batasan shred yang dijelaskan dalam peringatan di atas berlaku juga di sini.
> Perhatikan bahwa bash mencatat riwayat sesi saat ini ke disk saat keluar. Jika Anda keluar dari instans setelah menghapus `~/.bash_history`, lalu masuk kembali, Anda akan menemukan `~/.bash_history` telah dibuat ulang dan berisi semua perintah yang dijalankan selama sesi Anda sebelumnya.
> Program lain selain bash juga menulis riwayat ke disk, berhati-hatilah dan hapus atau kecualikan dot-file dan dot-directories yang tidak perlu.

- Membuat paketan dari instans berjalan memerlukan kunci privat dan sertifikat X.509 Anda. Simpan kredensial ini dan kredensial lainnya di lokasi yang tidak dipaketkan (misalnya penyimpanan instans).

# Memantau acara AMI menggunakan Amazon EventBridge

Ketika status Amazon Machine Image (AMI) berubah, Amazon EC2 menghasilkan peristiwa yang dikirim ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon Ev CloudWatch ents). Peristiwa dikirim ke bus EventBridge acara default dalam format JSON. Anda dapat menggunakan Amazon EventBridge untuk mendeteksi dan bereaksi terhadap peristiwa ini. Anda melakukan ini dengan membuat aturan EventBridge yang memicu tindakan sebagai respons terhadap suatu peristiwa. Misalnya, Anda dapat membuat EventBridge aturan yang mendeteksi kapan proses pembuatan AMI telah selesai dan kemudian memanggil topik Amazon SNS untuk mengirim pemberitahuan email kepada Anda.

Amazon EC2 menghasilkan `EC2 AMI State Change` peristiwa ketika AMI memasuki salah satu status berikut:

- `available`
- `failed`
- `deregistered`
- `disabled`

Peristiwa dihasilkan atas dasar upaya terbaik.

Tabel berikut ini mencantumkan operasi dan status AMI yang dapat dimasuki oleh AMI. Dalam tabel, Ya menunjukkan status yang dapat dimasuki AMI ketika operasi yang sesuai berjalan.

| Operasi AMI | available | failed | deregistered | disabled |
|---|---|---|---|---|
| CopyImage | Ya | Ya | | |
| CreateImage | Ya | Ya | | |
| CreateRestoreImageTask | Ya | Ya | | |

| Operasi AMI | available | failed | deregistered | disabled |
|---|---|---|---|---|
| DeregisterImage | | | Ya | |
| DisableImage | | | | Ya |
| EnableImage | Ya | | | |
| RegisterImage | Ya | Ya | | |

Peristiwa EC2 AMI State Change

- Detail peristiwa
- tersedia peristiwa
- gagal peristiwa
- dibatalkan pendaftaran peristiwa
- dinonaktifkan peristiwa

## Detail peristiwa

Anda dapat menggunakan bidang berikut dalam acara untuk membuat aturan yang memicu tindakan:

`"source": "aws.ec2"`

Mengidentifikasi bahwa peristiwa tersebut dari Amazon EC2.

`"detail-type": "EC2 AMI State Change"`

Mengidentifikasi nama peristiwa.

`"detail": { "ImageId": "ami-0abcdef1234567890", "State": "available", }`

Menyediakan ID AMI dan status AMI (`available`,`failed`,`deregistered`, atau`disabled`).

Untuk informasi selengkapnya, lihat berikut ini di Panduan EventBridge Pengguna Amazon:

- Acara EventBridge Amazon
- P EventBridge ola acara Amazon

- A EventBridge turan Amazon

Untuk tutorial tentang cara membuat fungsi Lambda dan EventBridge aturan yang menjalankan fungsi Lambda, lihat Tutorial: Mencatat status instans Amazon EC2 menggunakan EventBridge di Panduan Peng AWS Lambda embang.

## tersedia peristiwa

Berikut ini adalah contoh peristiwa yang dihasilkan Amazon EC2 saat AMI memasuki status `available` setelah keberhasilan operasi `CreateImage`, `CopyImage`, `RegisterImage`, `CreateRestoreImageTask`, atau `EnableImage`.

`"State": "available"` menunjukkan bahwa operasi berhasil.

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0abcdef1234567890"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0abcdef1234567890",
        "State": "available",
        "ErrorMessage": ""
    }
}
```

## gagal peristiwa

Berikut ini adalah contoh peristiwa yang dihasilkan Amazon EC2 saat AMI memasuki status `failed` setelah kegagalan operasi `CreateImage`, `CopyImage`, `RegisterImage`, atau `CreateRestoreImageTask`.

Bidang berikut memberikan informasi terkait:

- `"State": "failed"` – Menunjukkan bahwa operasi gagal.
- `"ErrorMessage": ""` – Memberikan alasan kegagalan operasi.

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0abcdef1234567890"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0abcdef1234567890",
        "State": "failed",
        "ErrorMessage": "Description of failure"
    }
}
```

## dibatalkan pendaftaran peristiwa

Berikut ini adalah contoh peristiwa yang dihasilkan Amazon EC2 saat AMI memasuki status `deregistered` setelah keberhasilan operasi `DeregisterImage`. Jika operasi gagal, tidak ada peristiwa yang dihasilkan. Kegagalan diketahui segera karena `DeregisterImage` merupakan operasi tersinkron.

`"State": "deregistered"` menunjukkan bahwa operasi `DeregisterImage` berhasil.

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0abcdef1234567890"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0abcdef1234567890",
        "State": "deregistered",
        "ErrorMessage": ""
    }
}
```

### dinonaktifkan peristiwa

Berikut ini adalah contoh peristiwa yang dihasilkan Amazon EC2 saat AMI memasuki status `disabled` setelah keberhasilan operasi `DisableImage`. Jika operasi gagal, tidak ada peristiwa yang dihasilkan. Kegagalan diketahui segera karena `DisableImage` merupakan operasi tersinkron.

`"State": "disabled"` menunjukkan bahwa operasi `DisableImage` berhasil.

```
{
    "version": "0",
    "id": "example-9f07-51db-246b-d8b8441bcdf0",
    "detail-type": "EC2 AMI State Change",
    "source": "aws.ec2",
    "account": "012345678901",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1::image/ami-0abcdef1234567890"],
    "detail": {
        "RequestId": "example-9dcc-40a6-aa77-7ce457d5442b",
        "ImageId": "ami-0abcdef1234567890",
        "State": "disabled",
        "ErrorMessage": ""
    }
}
```

## Memahami informasi penagihan AMI

Ada banyak Amazon Machine Image (AMI) yang dapat dipilih saat meluncurkan instans Anda, dan mereka mendukung berbagai platform dan fitur sistem operasi. Untuk memahami bagaimana AMI yang Anda pilih saat meluncurkan instans memengaruhi laba pada tag AWS ihan Anda, Anda dapat meneliti platform sistem operasi terkait dan informasi penagihan. Lakukan ini sebelum Anda meluncurkan Instans On-Demand atau Instans Spot, atau membeli Instans Cadangan.

Berikut adalah dua contoh bagaimana meneliti AMI Anda sebelumnya dapat membantu Anda memilih AMI yang paling sesuai dengan kebutuhan Anda:

- Untuk Instans Spot, Anda dapat menggunakan Detail platform AMI untuk mengonfirmasi bahwa AMI didukung untuk Instans Spot.
- Saat membeli Instans Cadangan, Anda dapat memastikan bahwa, Anda memilih platform sistem operasi (Platform), yang memetakan Detail platform pada AMI.

Untuk informasi selengkapnya tentang harga instans, lihat Harga Amazon EC2.

Daftar Isi


- Bidang informasi penagihan AMI
- Mencari detail penagihan dan penggunaan AMI
- Memverifikasi biaya AMI pada tagihan Anda


## Bidang informasi penagihan AMI

Bidang berikut menyediakan informasi penagihan yang terkait dengan AMI:

Detail platform AMI

Detail platform yang terkait akan dengan kode penagihan AMI. Sebagai contoh, `Red Hat Enterprise Linux`.

Operasi penggunaan

Operasi instans Amazon EC2 dan kode penagihan yang terkait dengan AMI. Misalnya, `RunInstances:0010`. Operasi penggunaan sesuai dengan lineitem/Operation kolom pada Laporan AWS Biaya dan Penggunaan (CUR) Anda dan di API Daftar AWS Harga.

Anda dapat melihat bidang ini di halaman Instans atau AMI di konsol Amazon EC2, atau dalam respons yang dikembalikan oleh deskripsi-gambar atau perintah Get-EC2Image.

### Contoh data: operasi penggunaan berdasarkan platform

Tabel berikut mencantumkan beberapa detail platform dan nilai operasi penggunaan yang dapat ditampilkan pada halaman Instans atau AMI di konsol Amazon EC2, atau dalam respons yang dikembalikan oleh deskripsi-gambar atau perintah Get-EC2Image.

| Detail platform | Operasi penggunaan [2] |
|---|---|
| Linux/UNIX | RunInstances |
| Red Hat BYOL Linux | RunInstances:00g0[3] |

| Detail platform | Operasi penggunaan [2] |
| --- | --- |
| Red Hat Enterprise Linux | RunInstances:0010 |
| Red Hat Enterprise Linux with HA | RunInstances:1010 |
| Red Hat Enterprise Linux with SQL Server Standard and HA | RunInstances:1014 |
| Red Hat Enterprise Linux with SQL Server Enterprise and HA | RunInstances:1110 |
| Red Hat Enterprise Linux with SQL Server Standard | RunInstances:0014 |
| Red Hat Enterprise Linux with SQL Server Web | RunInstances:0210 |
| Red Hat Enterprise Linux with SQL Server Enterprise | RunInstances:0110 |
| SQL Server Enterprise | RunInstances:0100 |
| SQL Server Standard | RunInstances:0004 |
| SQL Server Web | RunInstances:0200 |
| SUSE Linux | RunInstances:000g |
| Ubuntu Pro | RunInstances:0g00 |
| Windows | RunInstances:0002 |
| Windows BYOL | RunInstances:0800 |
| Windows with SQL Server Enterprise[1] | RunInstances:0102 |

| Detail platform | Operasi penggunaan [2] |
| --- | --- |
| Windows with SQL Server Standard[1] | RunInstances:0006 |
| Windows with SQL Server Web[1] | RunInstances:0202 |

[1] Jika dua lisensi perangkat lunak dikaitkan dengan AMI, bidang Detail Platform menampilkan keduanya.

[2] Jika Anda menjalankan Instans Spot, lineitem/Operation pada Laporan AWS Biaya dan Penggunaan Anda mungkin berbeda dari nilai operasi Penggunaan yang tercantum di sini. Misalnya, jika ditampilkan `lineitem/Operation`RunInstances:0010:SV006, itu berarti Amazon EC2 menjalankan Red Hat Enterprise Linux Spot Instance-hour di AS Timur (Virginia Utara) di Zona 6.

[3] Ini muncul seperti RunInstances (Linux/UNIX) dalam laporan penggunaan Anda.

## Mencari detail penagihan dan penggunaan AMI

Properti berikut dapat membantu Anda memverifikasi biaya AMI pada tagihan Anda:

- Detail platform
- Operasi penggunaan
- ID AMI

Console

Untuk menemukan informasi penagihan AMI untuk AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pada tab Detail, temukan Detail Platform dan Oper asi Penggunaan.

Untuk menemukan informasi penagihan AMI untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pada tab Detail, perluas Detail Instance dan temukan Detail Platform dan Oper asi Penggunaan.

AWS CLI

Untuk menemukan informasi penagihan AMI untuk AMI

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
    --query "Images[].
{PlatformDetails:PlatformDetails,UsageOperation:UsageOperation}"
```

Berikut ini adalah contoh output untuk Linux AMI.

```
[
    {
        "PlatformDetails": "Linux/UNIX",
        "UsageOperation": "RunInstances"
    }
]
```

Untuk menemukan informasi penagihan AMI untuk sebuah instance

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query "Reservations[].Instances[].
{PlatformDetails:PlatformDetails,UsageOperation:UsageOperation}"
```

Berikut ini adalah contoh output untuk instance Windows.

```
[
```

```
    {
        "PlatformDetails": "Windows",
        "UsageOperation": "RunInstances:0002"
    }
]
```

PowerShell

Untuk menemukan informasi penagihan AMI untuk AMI

Gunakan Get-EC2Image cmdlet.

```
Get-EC2Image `
    -ImageId ami-0abcdef1234567890 | `
    Format-List PlatformDetails, UsageOperation
```

Berikut ini adalah contoh output untuk Linux AMI.

```
PlatformDetails : Linux/UNIX
UsageOperation  : RunInstances
```

Untuk menemukan informasi penagihan AMI untuk sebuah instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances | `
    Format-List PlatformDetails, UsageOperation
```

Berikut ini adalah contoh output untuk instance Windows.

```
PlatformDetails : Windows
UsageOperation  : RunInstances:0002
```

## Memverifikasi biaya AMI pada tagihan Anda

Untuk memastikan bahwa Anda tidak menimbulkan biaya yang tidak direncanakan, Anda dapat memverifikasi bahwa informasi penagihan untuk instance di Laporan AWS Biaya dan Penggunaan

(CUR) Anda cocok dengan informasi penagihan yang terkait dengan AMI yang Anda gunakan untuk meluncurkan instans.

Untuk memverifikasi informasi penagihan, temukan ID instans di CUR Anda dan periksa nilai yang sesuai di kolom `lineitem/Operation`. Nilai harus sesuai dengan nilai untuk Operasi penggunaan yang berkaitan dengan AMI.

Sebagai contoh, `ami-0123456789EXAMPLE` AMI memiliki informasi penagihan berikut:

- Detail platform = `Red Hat Enterprise Linux`
- Operasi penggunaan = `RunInstances:0010`

Jika Anda meluncurkan instans menggunakan AMI ini, Anda dapat mencari ID instans di CUR dan memeriksa nilai yang sesuai di kolom `lineitem/Operation`. Dalam contoh ini, nilainya harus berupa `RunInstances:0010`.

# Kuota AMI di Amazon EC2

Kuota berikut berlaku untuk membuat dan berbagi AMIs. Kuota berlaku per Wilayah AWS.

| Nama kuota | Deskripsi | Kuota default per Wilayah |
|---|---|---|
| AMIs | Jumlah maksimum publik dan swasta yang AMIs diizinkan per Wilayah. Ini termasuk tersedia, tertunda, dan dinonaktifkan AMIs, dan AMIs di Recycle Bin. | 50.000 |
| Publik AMIs | Jumlah maksimum publik AMIs, termasuk publik AMIs di Recycle Bin, diperbolehkan per Wilayah. | 5 |
| Berbagi AMI | Jumlah maksimum entitas (organisasi, unit organisasi (OUs), dan akun) yang dapat dibagikan AMI di Wilayah. | 1.000 |

| Nama kuota | Deskripsi | Kuota default per Wilayah |
| --- | --- | --- |
| | Perhatikan bahwa jika Anda berbagi AMI dengan organisasi atau OU, jumlah akun dalam organisasi atau OU tidak dihitung dalam kuota. | |

Jika Anda melebihi kuota dan ingin membuat atau berbagi lebih banyak AMIs, Anda dapat melakukan hal berikut:

- Jika Anda melebihi total AMIs atau AMIs kuota publik, pertimbangkan untuk membatalkan pendaftaran gambar yang tidak digunakan.
- Jika Anda melebihi AMIs kuota publik Anda, pertimbangkan untuk membuat satu atau lebih publik AMIs pribadi.
- Jika Anda melebihi kuota berbagi AMI, pertimbangkan untuk berbagi AMIs dengan organisasi atau OU, bukan akun terpisah.
- Minta kenaikan kuota untuk AMIs.

## Minta kenaikan kuota untuk AMIs

Jika Anda membutuhkan lebih dari kuota default AMIs, Anda dapat meminta kenaikan kuota.

Untuk meminta kenaikan kuota untuk AMIs

1. Buka konsol Service Quotas di https://console.aws.amazon.com/servicequotas/.
2. Di panel navigasi, pilih Layanan AWS .
3. Pilih Amazon Elastic Compute Cloud (Amazon EC2) dari daftar, atau ketik nama layanan di kotak pencarian.
4. Pilih kuota AMI untuk meminta kenaikan. Kuota AMI yang dapat Anda pilih adalah:
   - AMIs
   - Publik AMIs
   - Berbagi AMI
5. Pilih Ajukan peningkatan kuota.

6. Untuk Mengubah nilai kuota, masukkan nilai kuota yang baru, lalu pilih Ajukan.

Untuk melihat permintaan yang tertunda atau baru diselesaikan, pilih Dasbor dari panel navigasi. Untuk permintaan yang tertunda, pilih status permintaan untuk membuka penerimaan permintaan. Status awal dari permintaan adalah Tertunda. Setelah status berubah menjadi Kuota yang diminta, Anda akan melihat nomor kasus di bagian Nomor kasus Pusat Dukungan. Pilih nomor kasus untuk membuka tiket untuk permintaan Anda.

Setelah permintaan diselesaikan, Nilai kuota yang diterapkan untuk kuota tersebut diatur ke nilai baru.

Untuk informasi lebih lanjut, lihat Panduan Pengguna Kuota Layanan.

# Instans Amazon EC2

Instans Amazon EC2 adalah server virtual di lingkungan AWS cloud. Anda memiliki kontrol penuh atas instans Anda, dari saat pertama kali memulainya (disebut sebagai meluncurkan instance) hingga Anda menghapusnya (disebut sebagai penghentian instance). Anda dapat memilih dari berbagai sistem operasi saat meluncurkan instans Anda. Anda dapat terhubung ke instance Anda dan menyesuaikannya untuk memenuhi kebutuhan Anda. Misalnya, Anda dapat mengonfigurasi sistem operasi, menginstal pembaruan sistem operasi, dan menginstal aplikasi pada instans Anda.

Amazon EC2 menyediakan berbagai jenis instans. Anda dapat memilih jenis instans yang menyediakan sumber daya komputasi, memori, penyimpanan, dan kinerja jaringan yang Anda butuhkan untuk menjalankan aplikasi Anda.

Dengan Amazon EC2, Anda hanya membayar untuk apa yang Anda gunakan. Penagihan untuk instans dimulai saat Anda meluncurkan instans dan beralih ke status berjalan. Penagihan berhenti saat Anda menghentikan instans dan dilanjutkan saat Anda memulai instans. Saat Anda mengakhiri instans, penagihan akan berhenti ketika instans beralih ke status shutdown.

Amazon EC2 menyediakan fitur yang dapat Anda gunakan untuk mengoptimalkan kinerja dan biaya instans Anda. Misalnya, Anda dapat menggunakan Amazon EC2 Fleet atau Amazon EC2 Auto Scaling untuk meningkatkan atau menurunkan kapasitas saat penggunaan instans berubah. Anda dapat mengurangi biaya instans menggunakan Instans Spot atau Paket Tabungan.

Instans ter  kelola dikelola oleh penyedia layanan, seperti Mode Otomatis Amazon EKS.  Anda tidak dapat mengubah pengaturan instans terkelola secara langsung. Instans terkelola diidentifikasi dengan  nilai  sebenarnya di  bidang  Terkelola. Untuk informasi selengkapnya, lihat Instans terkelola Amazon EC2.

Fitur dan tugas

- Tipe instans Amazon EC2
- Instans terkelola Amazon EC2
- Gunakan virtualisasi bersarang untuk menjalankan hypervisor di instans Amazon EC2
- Opsi penagihan dan pembelian Amazon EC2
- Simpan parameter peluncuran instans di templat peluncuran Amazon EC2
- Luncurkan instans Amazon EC2

- Sambungkan ke instans EC2
- Perubahan status instans Amazon EC2
- Pemulihan instans otomatis
- Menggunakan metadata instans untuk mengelola instans EC2 Anda
- Mendeteksi apakah host adalah instance EC2
- Dokumen identitas instans untuk instans Amazon EC2
- Kepatuhan STIG untuk instans EC2 Anda
- Sinkronisasi jam dan waktu presisi pada instans EC2 Anda
- Manajer Kapasitas EC2
- Mengelola driver perangkat untuk instans EC2
- Konfigurasikan instans Windows Amazon EC2 Anda
- Tingkatkan instance EC2 Windows ke versi Windows Server yang lebih baru
- Tutorial: Hubungkan instans Amazon EC2 ke basis data Amazon RDS

# Tipe instans Amazon EC2

Saat meluncurkan sebuah instans, tipe instans yang Anda pilih menentukan perangkat keras komputer host yang digunakan untuk instans Anda. Setiap tipe instans menawarkan kemampuan komputasi, memori, dan penyimpanan yang berbeda, serta dikelompokkan dalam sebuah keluarga instans berdasarkan kemampuan tersebut. Pilih tipe instans berdasarkan persyaratan aplikasi atau perangkat lunak yang rencananya akan Anda jalankan pada instans Anda. Untuk informasi selengkapnya tentang fitur dan kasus penggunaan, lihat Jenis Instans Amazon EC2.

Amazon EC2 mengkhususkan beberapa sumber daya komputer host, seperti CPU, memori, dan penyimpanan instans, untuk instans tertentu. Amazon EC2 berbagi sumber daya lain dari komputer host, seperti jaringan dan subsistem disk, antarinstans. Jika setiap instans pada komputer host mencoba menggunakan sebanyak mungkin salah satu sumber daya bersama ini, masing-masing menerima bagian yang sama dari sumber daya tersebut. Namun, jika sumber daya kurang digunakan, sebuah instans dapat menggunakan bagian yang lebih tinggi dari sumber daya tersebut selama tersedia.

Setiap tipe instans memberikan performa minimum yang lebih tinggi atau lebih rendah dari sumber daya bersama. Misalnya, jenis instans dengan I/O kinerja tinggi memiliki alokasi sumber daya

bersama yang lebih besar. Mengalokasikan bagian yang lebih besar dari sumber daya bersama juga mengurangi varians I/O kinerja. Untuk sebagian besar aplikasi, I/O kinerja moderat lebih dari cukup. Namun, untuk aplikasi yang memerlukan I/O kinerja yang lebih besar atau lebih konsisten, pertimbangkan jenis instance dengan I/O kinerja yang lebih tinggi.

Daftar Isi


- Tipe instans yang tersedia
- Spesifikasi perangkat keras
- Jenis hypervisor
- Tipe virtualisasi AMI
- Prosesor
- Menemukan tipe instans Amazon EC2
- Dapatkan rekomendasi dari pencari tipe instans EC2
- Dapatkan rekomendasi instans EC2 dari Compute Optimizer
- Perubahan jenis instans Amazon EC2
- Instance performa yang dapat melonjak
- Akselerasi kinerja dengan instans GPU
- Instans Mac Amazon EC2
- Jenis EBS-optimized instans Amazon
- Opsi CPU untuk instans Amazon EC2
- AMD SEV-SNP untuk instans Amazon EC2
- Kontrol status prosesor untuk instans Amazon EC2 Linux


## Tipe instans yang tersedia

Amazon EC2 menyediakan berbagai pilihan tipe instans yang dioptimalkan agar sesuai dengan berbagai kasus penggunaan. Tipe instans terdiri dari berbagai kombinasi CPU, memori, penyimpanan, dan kapasitas jaringan, serta memberi Anda fleksibilitas untuk memilih campuran sumber daya yang sesuai untuk aplikasi Anda. Setiap tipe instans menyertakan satu atau beberapa ukuran instans, memungkinkan Anda untuk menskalakan sumber daya sesuai dengan persyaratan beban kerja target Anda.

Konvensi penamaan tipe instance

Nama didasarkan pada keluarga instans, generasi, keluarga prosesor, kemampuan, dan ukuran. Untuk informasi selengkapnya, lihat Konvensi penamaan di Panduan Jenis Instans Amazon EC2.

Menemukan tipe instans

Untuk menentukan jenis instans mana yang memenuhi persyaratan Anda, seperti Wilayah yang didukung, sumber daya komputasi, atau sumber daya penyimpanan, lihat Menemukan tipe instans Amazon EC2 dan spesifikasi jenis instans Amazon EC2 di Panduan Jenis Instans Amazon EC2.

## Spesifikasi perangkat keras

Untuk spesifikasi tipe instans terperinci, lihat Spesifikasi di Panduan Jenis Instans Amazon EC2. Untuk informasi harga, lihat Harga Amazon EC2 On-Demand .

Untuk menentukan tipe instans yang paling sesuai dengan kebutuhan Anda, kami menyarankan Anda meluncurkan sebuah instans dan menggunakan aplikasi tolok ukur Anda sendiri. Karena Anda membayar dengan basis per detik instans, akan lebih mudah dan murah untuk menguji banyak tipe instans sebelum membuat keputusan. Jika kebutuhan berubah, bahkan setelah membuat keputusan, Anda dapat mengubah tipe instans nanti. Untuk informasi selengkapnya, lihat Perubahan jenis instans Amazon EC2.

## Jenis hypervisor

Amazon EC2 mendukung hypervisor berikut: Xen dan Nitro.

Nitro-based contoh

- Tujuan umum: M5 | M5a | M5ad | M5d | M5dn | M5n | M5Zn | M6a | M6g | M6gd | M6i | M6id | M6idn | M6in | M7a | M7g | M7gd | M7i | | M8a | M8azN | M8g | M8gb | Gd | M8gn | M8i | M8id | | M8in | M8idn | M8iN | M8iB | M8iDB | M9g | M9gd | T3 | T3a | M7i-flex T4g M8i-flex
- Komputasi dioptimalkan: C5 | C5a | C5ad | C5d | C5n | C6a | C6g | C6gD | C6gN | C6i | C6id | C6in | C7a | C7g | C7gD | C7gN | C7i | | C8a | C8g | C8GB | C8gD | C8gD | C8gD | C8gD | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gN | C8gi | C8id | | C8in | C8in | C8iB | C9g | C9gD C7i-flex C8i-flex
- Memori dioptimalkan: R5 | R5a | R5ad | R5b | R5d | R5dn | R5n | R6a | R6g | R6gd | R6i | R6id | R6idn | R6in | R7a | R7g | R7gd | R7i | R7iz | R8a | R8g | R8g | R8Gd | R8gn | R8i | R8id | | R8in | R8idn | R8iB | R8iDB | R9g | R9gD | | | | | | | | | | | | X2gd | x2iEDn | x2ieDN | x8g | x8aEDz | x8i | z1d R8i-flex U-3tb1 U-6tb1 U-9tb1 U-12tb1 U-18tb1 U-24tb1 U7i-6tb U7i-8tb U7i-12tb U7in-16tb U7in-24tb U7in-32tb U7inh-32tb

- Penyimpanan dioptimalkan:  D3 | D3en | i3EN | i4G | i4i | i7i | i7iE | i8G | i8GE | i4GN | IS4Gen
- Komputasi yang dipercepat:  DL1 | DL2q | F2 | G4ad | G4dn | G5 | G5g | G6 | G6e | G6f | Gr6 | GR6f | G7 | G7e | Inf1 | Inf2 | P4d | P4de | P5 | P5e | P5en | | | Trn1 | Trn1n | Trn2 | Trn2u | VT1 P6-B200 P6-B300 P6e-GB200
- High-performance komputasi:  HPC6a | HPC6id | HPC7a | HPC7g | HPC8a
- Generasi sebelumnya:  A1 | P3dn

Untuk informasi selengkapnya tentang versi Nitro hypervisor yang didukung, lihat Dukungan fitur Jaringan  di Panduan Jenis Instans  Amazon EC2.

Xen-based contoh

- Tujuan umum: M1 | M2 | M3 | M4 | T1 | T2
- Komputasi dioptimalkan: C1 | C3 | C4
- Memori dioptimalkan: R3 | R4 | X1 | X1e
- Penyimpanan dioptimalkan: D2 | H1 | I2 | I3
- Komputasi yang dipercepat: G3 | P3

## Tipe virtualisasi AMI

Tipe virtualisasi instans Anda ditentukan oleh AMI yang Anda gunakan untuk meluncurkannya. Tipe instans generasi saat ini hanya mendukung mesin virtual perangkat keras (HVM). Beberapa jenis instans generasi sebelumnya mendukung paravirtual (PV) dan beberapa AWS Regions mendukung instans PV. Untuk informasi selengkapnya, lihat Tipe virtualisasi.

Untuk mendapatkan performa terbaik, kami menyarankan Anda menggunakan AMI HVM. Selain itu, AMI HVM diperlukan untuk memanfaatkan jaringan yang ditingkatkan. Virtualisasi HVM menggunakan teknologi bantuan perangkat keras yang disediakan oleh platform. AWS Dengan virtualisasi HVM, VM tamu berjalan seolah-olah berada di platform perangkat keras native, hanya saja VM tersebut masih menggunakan jaringan PV dan driver penyimpanan untuk meningkatkan performanya.

## Prosesor

Instans EC2 mendukung berbagai prosesor.

Prosesor

- Prosesor Intel
- Prosesor AMD
- AWS Prosesor Graviton
- AWS Trainium
- AWS Inferentia

## Prosesor Intel

Instans Amazon EC2 yang berjalan pada prosesor Intel mungkin menyertakan fitur prosesor berikut. Tidak semua instance yang berjalan pada prosesor Intel mendukung semua fitur prosesor ini. Untuk informasi tentang fitur yang tersedia untuk setiap jenis instans, lihat Jenis Instans Amazon EC2.

- Intel AES New Instructions (AES-NI) — Set instruksi AES-NI enkripsi Intel meningkatkan algoritma Advanced Encryption Standard (AES) asli untuk memberikan perlindungan data yang lebih cepat dan keamanan yang lebih baik. Semua instans EC2 generasi saat ini mendukung fitur prosesor ini.
- Intel Advanced Vector Extensions (Intel AVX, Intel AVX2, dan IntelAVX-512) — Intel AVX dan Intel AVX2 berukuran 256-bit, dan Intel AVX-512 adalah ekstensi set instruksi 512-bit yang dirancang untuk aplikasi yang intensif Floating Point (FP). Instruksi Intel AVX meningkatkan kinerja untuk aplikasi seperti gambar dan audio/video pemrosesan, simulasi ilmiah, analitik keuangan, dan pemodelan dan analisis 3D. Fitur-fitur ini hanya tersedia pada instans yang diluncurkan dengan AMI HVM.
- Teknologi Intel Turbo Boost — Prosesor Intel Turbo Boost Technology secara otomatis menjalankan inti lebih cepat dari frekuensi operasi dasar.
- Intel Deep Learning Boost (Intel DL Boost) — Mempercepat kasus penggunaan AI deep learning. Prosesor Intel Xeon Scalable Generasi ke-2 memperluas Intel AVX-512 dengan Vector Neural Network Instruction (VNNI/INT8) baru yang secara signifikan meningkatkan kinerja inferensi pembelajaran mendalam dibandingkan prosesor Intel Xeon Scalable generasi sebelumnya (dengan FP32) untuk gambar recognition/segmentation, deteksi objek, pengenalan suara, terjemahan bahasa, sistem rekomendasi, pembelajaran penguatan, dan banyak lagi. VNNI mungkin tidak kompatibel dengan semua distribusi Linux.

  Instans berikut mendukung VNNI: `M5n`, `R5n`, `M5dn`, `M5zn`, `R5b`, `R5dn`, `D3`, `D3en`, dan `C6i`. Instans `C5` dan `C5d` mendukung VNNI hanya untuk instans `12xlarge`, `24xlarge`, dan `metal`.

Kebingungan dapat terjadi akibat konvensi penamaan industri untuk CPU 64-bit. Produsen chip Advanced Micro Devices (AMD) memperkenalkan arsitektur 64-bit pertama yang sukses secara komersial berbasis set instruksi Intel x86. Akibatnya, arsitektur tersebut secara luas disebut sebagai AMD64, terlepas dari produsen chip-nya. Windows dan beberapa distribusi Linux mengikuti praktik ini. Hal ini menjelaskan alasan informasi sistem internal pada instans yang menjalankan Ubuntu atau Windows menampilkan arsitektur CPU sebagai AMD64 meskipun instans tersebut berjalan pada perangkat keras Intel.

## Prosesor AMD

Instans Amazon EC2 yang berjalan pada prosesor AMD EPYC dapat membantu Anda mengoptimalkan biaya dan kinerja untuk beban kerja Anda. Instans ini mungkin mendukung fitur prosesor berikut. Tidak semua instance yang berjalan pada prosesor AMD mendukung semua fitur prosesor ini. Untuk informasi tentang fitur yang tersedia untuk setiap jenis instans, lihat Jenis Instans Amazon EC2.

- Enkripsi Memori Aman AMD (SME)
- Enkripsi Memori Kunci Tunggal Transparan AMD (TSME)
- Ekstensi Vektor Lanjutan AMD (AVX)
- Paging Bersarang Ter Virtualization-Secure enkripsi AMD Aman () SEV-SNP
- Instruksi Jaringan Saraf Vektor (VNNI)
- BFLoAT16

## AWS Prosesor Graviton

AWS Graviton adalah rangkaian prosesor yang dirancang untuk memberikan kinerja harga terbaik untuk beban kerja Anda yang berjalan di instans Amazon EC2.

Untuk informasi selengkapnya, lihat Memulai dengan Graviton.

## AWS Trainium

Instans yang didukung oleh AWS Trainium dirancang khusus untuk pelatihan pembelajaran mendalam berkinerja tinggi dan hemat biaya. Anda dapat menggunakan instance ini untuk melatih pemrosesan bahasa alami, visi komputer, dan model rekomendasi yang digunakan di berbagai aplikasi, seperti pengenalan suara, rekomendasi, deteksi penipuan, dan klasifikasi gambar dan video. Gunakan alur kerja Anda yang ada dalam kerangka kerja ML populer, seperti PyTorch dan TensorFlow.

### AWS Inferentia

Instans yang didukung oleh AWS Inferentia dirancang untuk mempercepat pembelajaran mesin. Mereka memberikan inferensi pembelajaran mesin kinerja tinggi dan latensi rendah. Instans ini dioptimalkan untuk menerapkan model deep learning (DL) untuk aplikasi, seperti pemrosesan bahasa alami, deteksi dan klasifikasi objek, personalisasi dan pemfilteran konten, dan pengenalan ucapan.

Ada berbagai cara untuk memulai:

- Gunakan SageMaker AI, layanan yang dikelola sepenuhnya yang merupakan cara termudah untuk memulai dengan model pembelajaran mesin. Untuk informasi selengkapnya, lihat Mem ulai dengan SageMaker AI di Panduan Pengembang SageMaker AI Amazon.
- Luncurkan instans Inf1 atau Inf2 menggunakan AMI Deep Learning. Untuk informasi selengkapnya, lihat AWS Inferentia dengan DLAMI di Panduan Developer AWS Deep Learning AMIs .
- Luncurkan instans Inf1 atau Inf2 menggunakan AMI Anda sendiri dan instal AWS Neuron SDK, yang memungkinkan Anda untuk mengompilasi, menjalankan, dan membuat profil model deep learning untuk AWS Inferentia.
- Luncurkan instance kontainer menggunakan instans Inf1 atau Inf2 dan Amazon ECS-optimized AMI. Untuk informasi selengkapnya, lihat AMI Amazon Linux 2 (Inferentia) di Panduan Developer Amazon Elastic Container Service.
- Buat klaster Amazon EKS dengan simpul yang menjalankan instans Inf1. Untuk informasi selengkapnya, lihat dukungan Inferentia di Panduan Pengguna Amazon EKS.

## Menemukan tipe instans Amazon EC2

Sebelum dapat meluncurkan sebuah instans, Anda harus memilih tipe instans yang akan digunakan. Tipe instans yang Anda pilih mungkin bergantung pada sumber daya yang dibutuhkan oleh beban kerja Anda, seperti sumber daya komputasi, memori, atau penyimpanan. Akan bermanfaat untuk mengidentifikasi beberapa tipe instans yang mungkin sesuai dengan beban kerja Anda dan mengevaluasi kinerjanya di lingkungan pengujian. Tidak ada pengganti untuk mengukur performa aplikasi Anda di bawah beban.

Anda bisa mendapatkan saran dan panduan untuk jenis instans EC2 menggunakan pencari tipe instans EC2. Untuk informasi selengkapnya, lihat the section called “Pencari tipe instans EC2”.

Jika Anda sudah menjalankan instans EC2, Anda dapat menggunakannya AWS Compute Optimizer untuk mendapatkan rekomendasi tentang jenis instans yang harus Anda gunakan untuk

meningkatkan kinerja, menghemat uang, atau keduanya. Untuk informasi selengkapnya, lihat the section called “Rekomendasi Compute Optimizer”.

Tugas

- Untuk menemukan tipe instans menggunakan konsol
- Jelaskan jenis instance menggunakan AWS CLI
- Menemukan jenis instance menggunakan AWS CLI
- Menemukan jenis instance menggunakan Tools for PowerShell

## Untuk menemukan tipe instans menggunakan konsol

Anda dapat menemukan tipe instans yang memenuhi kebutuhan Anda menggunakan konsol Amazon EC2.

Untuk menemukan tipe instans menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi, pilih Wilayah untuk meluncurkan instans Anda. Anda dapat memilih Wilayah yang tersedia untuk Anda, terlepas dari lokasi Anda.
3. Pada panel navigasi, pilih Tipe Instans.
4. (Opsional) Pilih ikon preferensi (roda gigi) untuk memilih atribut jenis instans yang akan ditampilkan, seperti harga  On-Demand Linux, lalu pilih Kon  firmasi. Atau, pilih nama tipe instans untuk membuka halaman detailnya dan melihat semua atribut yang tersedia melalui konsol. Konsol tidak menampilkan semua atribut yang tersedia melalui API atau baris perintah.
5. Gunakan atribut tipe instans untuk memfilter daftar tipe instans yang ditampilkan ke hanya tipe instans yang memenuhi kebutuhan Anda. Misalnya, Anda dapat memfilter atribut berikut:
   - Zona ketersediaan — Nama Zona Ketersediaan, Local Zone, atau Wavelength Zone. Untuk informasi selengkapnya, lihat the section called “Wilayah dan Zona”.
   - VCPU atau Inti — Jumlah vCPU atau inti.
   - Memori (GiB) — Ukuran memori, dalam GiB.
   - Performa jaringan – Performa jaringan, dalam Gigabits.
   - Penyimpanan instans lokal – Menunjukkan apakah tipe instans memiliki penyimpanan instans lokal (`true` | `false`).

6. (Opsional) Untuk membandingkan secara langsung, pilih kotak centang untuk beberapa tipe instans. Perbandingan ditampilkan di bagian bawah layar.
7. (Opsional) Untuk menyimpan daftar tipe instans ke file nilai dipisahkan koma (.csv) untuk peninjauan lebih lanjut, pilih Tindakan, Unduh CSV daftar. File tersebut mencakup semua tipe instans yang cocok dengan filter yang Anda atur.
8. (Opsional) Untuk meluncurkan instans menggunakan tipe instans yang sesuai dengan kebutuhan Anda, pilih kotak centang untuk tipe instans dan pilih Tindakan, Luncurkan instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

## Jelaskan jenis instance menggunakan AWS CLI

Anda dapat menggunakan perintah describe-instance-types untuk menggambarkan jenis instance tertentu.

Untuk mendeskripsikan tipe instance secara lengkap

Perintah berikut menampilkan semua detail yang tersedia untuk jenis instance yang ditentukan. Outputnya panjang, jadi dihilangkan di sini.

```
aws ec2 describe-instance-types \
    --instance-types t2.micro \
    --region us-east-2
```

Menjelaskan jenis instance dan memfilter output

Perintah berikut menampilkan detail jaringan untuk jenis instance yang ditentukan.

```
aws ec2 describe-instance-types \
    --instance-types t2.micro \
    --region us-east-2 \
    --query "InstanceTypes[].NetworkInfo"
```

Berikut ini adalah output contoh.

```
[
    {
        "NetworkPerformance": "Low to Moderate",
        "MaximumNetworkInterfaces": 2,
        "MaximumNetworkCards": 1,
```

```
        "DefaultNetworkCardIndex": 0,
        "NetworkCards": [
            {
                "NetworkCardIndex": 0,
                "NetworkPerformance": "Low to Moderate",
                "MaximumNetworkInterfaces": 2,
                "BaselineBandwidthInGbps": 0.064,
                "PeakBandwidthInGbps": 1.024
            }
        ],
        "Ipv4AddressesPerInterface": 2,
        "Ipv6AddressesPerInterface": 2,
        "Ipv6Supported": true,
        "EnaSupport": "unsupported",
        "EfaSupported": false,
        "EncryptionInTransitSupported": false,
        "EnaSrdSupported": false
    }
]
```

Perintah berikut menampilkan memori yang tersedia untuk jenis instance yang ditentukan.

```
aws ec2 describe-instance-types \
    --instance-types t2.micro \
    --region us-east-2 \
    --query "InstanceTypes[].MemoryInfo"
```

Berikut ini adalah output contoh.

```
[
    {
        "SizeInMiB": 1024
    }
]
```

## Menemukan jenis instance menggunakan AWS CLI

Anda dapat menggunakan perintah describe-instance-types dan describe-instance-type-offers untuk menemukan jenis instance yang memenuhi kebutuhan Anda.

### Contoh

- Contoh 1: Menemukan jenis instans berdasarkan Availability Zone

- Contoh 2: Temukan jenis instance berdasarkan ukuran memori yang tersedia
- Contoh 3: Menemukan jenis instans berdasarkan penyimpanan instans yang tersedia
- Contoh 4: Menemukan jenis instance yang mendukung hibernasi


Contoh 1: Menemukan jenis instans berdasarkan Availability Zone

Contoh berikut hanya menampilkan jenis instans yang ditawarkan di Zona Ketersediaan yang ditentukan.

```
aws ec2 describe-instance-type-offerings \
    --location-type "availability-zone" \
    --filters "Name=location,Values=us-east-2a" \
    --region us-east-2 \
    --query "InstanceTypeOfferings[*].[InstanceType]" --output text | sort
```

Outputnya adalah daftar jenis instance, diurutkan berdasarkan abjad. Berikut ini adalah awal dari output saja.

```
a1.2xlarge
a1.4xlarge
a1.large
a1.medium
a1.metal
a1.xlarge
c4.2xlarge
   ...
```

Contoh 2: Temukan jenis instance berdasarkan ukuran memori yang tersedia

Contoh berikut hanya menampilkan jenis instance generasi saat ini dengan memori 64 GiB (65536 MiB).

```
aws ec2 describe-instance-types \
    --filters "Name=current-generation,Values=true" "Name=memory-info.size-in-
mib,Values=65536" \
    --region us-east-2 \
    --query "InstanceTypes[*].[InstanceType]" --output text | sort
```

Outputnya adalah daftar jenis instance, diurutkan berdasarkan abjad. Berikut ini adalah awal dari output saja.

```
c5a.8xlarge
c5ad.8xlarge
c6a.8xlarge
c6g.8xlarge
c6gd.8xlarge
c6gn.8xlarge
c6i.8xlarge
c6id.8xlarge
c6in.8xlarge
   ...
```

Contoh 3: Menemukan jenis instans berdasarkan penyimpanan instans yang tersedia

Contoh berikut menampilkan ukuran total penyimpanan instans untuk semua instans R7 dengan volume penyimpanan instans.

```
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=r7*" "Name=instance-storage-
supported,Values=true" \
    --region us-east-2 \
    --query "InstanceTypes[].[InstanceType, InstanceStorageInfo.TotalSizeInGB]" \
    --output table
```

Berikut ini adalah output contoh.

```
---------------------------
|  DescribeInstanceTypes  |
+----------------+--------+
|  r7gd.xlarge   |  237   |
|  r7gd.8xlarge  |  1900  |
|  r7gd.16xlarge |  3800  |
|  r7gd.medium   |  59    |
|  r7gd.4xlarge  |  950   |
|  r7gd.2xlarge  |  474   |
|  r7gd.metal    |  3800  |
|  r7gd.large    |  118   |
|  r7gd.12xlarge |  2850  |
+----------------+--------+
```

Contoh 4: Menemukan jenis instance yang mendukung hibernasi

Contoh berikut menampilkan jenis instance yang mendukung hibernasi.

```
aws ec2 describe-instance-types \
    --filters "Name=hibernation-supported,Values=true" \
    --region us-east-2 \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

Outputnya adalah daftar jenis instance, diurutkan berdasarkan abjad. Berikut ini adalah awal dari output saja.

```
c4.2xlarge
c4.4xlarge
c4.8xlarge
c4.large
c4.xlarge
c5.12xlarge
c5.18xlarge
c5.2xlarge
c5.4xlarge
c5.9xlarge
...
```

### Menemukan jenis instance menggunakan Tools for PowerShell

Anda dapat menggunakan Get-EC2InstanceTypeOffering cmdlet Get-EC2InstanceType and untuk menemukan jenis instans yang memenuhi kebutuhan Anda.

Contoh

- Menemukan jenis instans berdasarkan Availability Zone
- Temukan jenis instance berdasarkan ukuran memori yang tersedia
- Menemukan jenis instans berdasarkan penyimpanan instans yang tersedia
- Menemukan jenis instance yang mendukung hibernasi

#### Menemukan jenis instans berdasarkan Availability Zone

Contoh berikut hanya menampilkan jenis instans yang ditawarkan di Zona Ketersediaan yang ditentukan.

```
(Get-EC2InstanceTypeOffering `
    -LocationType "availability-zone" `
```

```
    -Region us-east-2 `
    -Filter @{Name="location"; Values="us-east-2a"}).InstanceType | Sort-Object `
```

Temukan jenis instance berdasarkan ukuran memori yang tersedia

Contoh berikut hanya menampilkan jenis instance generasi saat ini dengan memori 64 GiB (65536 MiB).

```
(Get-EC2InstanceType `
    -Filter @{Name="current-generation"; Values="true"},
            @{Name="memory-info.size-in-mib"; Values="65536"}).InstanceType | Sort-
Object
```

Menemukan jenis instans berdasarkan penyimpanan instans yang tersedia

Contoh berikut menampilkan ukuran total penyimpanan instans untuk semua instans R7 dengan volume penyimpanan instans.

```
Get-EC2InstanceType `
    -Filter @{Name="instance-type"; Values="r7*"},
            @{Name="instance-storage-supported"; Values="true"} | `
     Select InstanceType, @{Name="TotalSizeInGB";
 Expression={($_.InstanceStorageInfo.TotalSizeInGB)}}
```

Berikut ini adalah output contoh.

```
InstanceType  TotalSizeInGB
------------  -------------
r7gd.8xlarge           1900
r7gd.16xlarge          3800
r7gd.xlarge             237
r7gd.4xlarge            950
r7gd.medium              59
r7gd.2xlarge            474
r7gd.large              118
r7gd.metal             3800
r7gd.12xlarge          2850
```

Menemukan jenis instance yang mendukung hibernasi

Contoh berikut menampilkan jenis instance yang mendukung hibernasi.

```
(Get-EC2InstanceType `
    -Filter @{Name="hibernation-supported"; Values="true"}).InstanceType | Sort-Object
```

## Dapatkan rekomendasi dari pencari tipe instans EC2

Pencari tipe instans EC2 mempertimbangkan kasus penggunaan, jenis beban kerja, preferensi produsen CPU, dan cara Anda memprioritaskan harga dan kinerja, serta parameter tambahan yang dapat Anda tentukan. Kemudian menggunakan data ini untuk memberikan saran dan panduan untuk jenis instans Amazon EC2 yang paling sesuai dengan beban kerja baru Anda.

Dengan begitu banyak jenis instans yang tersedia, menemukan jenis instans yang tepat untuk beban kerja Anda bisa memakan waktu dan rumit. Dengan menggunakan pencari tipe instans EC2, Anda dapat tetap mendapatkan informasi terbaru tentang jenis instans terbaru dan mencapai kinerja harga terbaik untuk beban kerja Anda.

Anda bisa mendapatkan saran dan panduan untuk jenis instans EC2 menggunakan konsol Amazon EC2. Anda juga dapat pergi langsung ke Amazon Q untuk meminta saran tipe instans. Untuk informasi selengkapnya, lihat Panduan Pengguna Pengembang Amazon Q.

Jika Anda mencari rekomendasi tipe instans untuk beban kerja yang ada, gunakan AWS Compute Optimizer. Untuk informasi selengkapnya, lihat Dapatkan rekomendasi instans EC2 dari Compute Optimizer.

### Gunakan pencari tipe instans EC2

Di konsol Amazon EC2, Anda bisa mendapatkan saran jenis instans dari pencari tipe instans EC2 di wizard instance peluncuran, saat membuat template peluncuran, atau di halaman Jenis instans.

Gunakan petunjuk berikut untuk mendapatkan saran dan panduan untuk jenis instans EC2 menggunakan pencari tipe instans EC2 di konsol Amazon EC2. Untuk melihat animasi langkah-langkahnya, lihatMelihat animasi: Dapatkan saran jenis instans menggunakan pencari tipe instans EC2.

Untuk mendapatkan saran jenis instans menggunakan pencari tipe instans EC2

1. Mulai proses Anda menggunakan salah satu dari berikut ini:
   - Ikuti prosedur untuk meluncurkan instans. Di samping Tipe instans, pilih tautan Dapatkan saran.

- Ikuti prosedur untuk membuat template peluncuran. Di samping Tipe instans, pilih tautan Dapatkan saran.
- Di panel navigasi, pilih Jenis Instans, lalu pilih tombol pencari jenis instans.

2. Di layar Dapatkan saran tentang pemilihan jenis instans, lakukan hal berikut:

   a. Tentukan persyaratan jenis instans Anda dengan memilih opsi untuk Jenis beban kerja, Kas us penggunaan, Priorit as, dan produsen CPU.

   b. (Opsional) Untuk menentukan persyaratan yang lebih rinci untuk beban kerja Anda, lakukan hal berikut:

      i. Perluas parameter lanjutan.

      ii. Untuk menambahkan parameter, pilih parameter, pilih Tam bah, dan tentukan nilai untuk parameter. Ulangi untuk setiap parameter tambahan yang ingin Anda tambahkan. Untuk menunjukkan tidak ada nilai minimum atau maksimum, biarkan bidang kosong.

      iii. Untuk menghapus parameter setelah menambahkannya, pilih X di sebelah parameter.

   c. Pilih Dapatkan saran tipe instans.

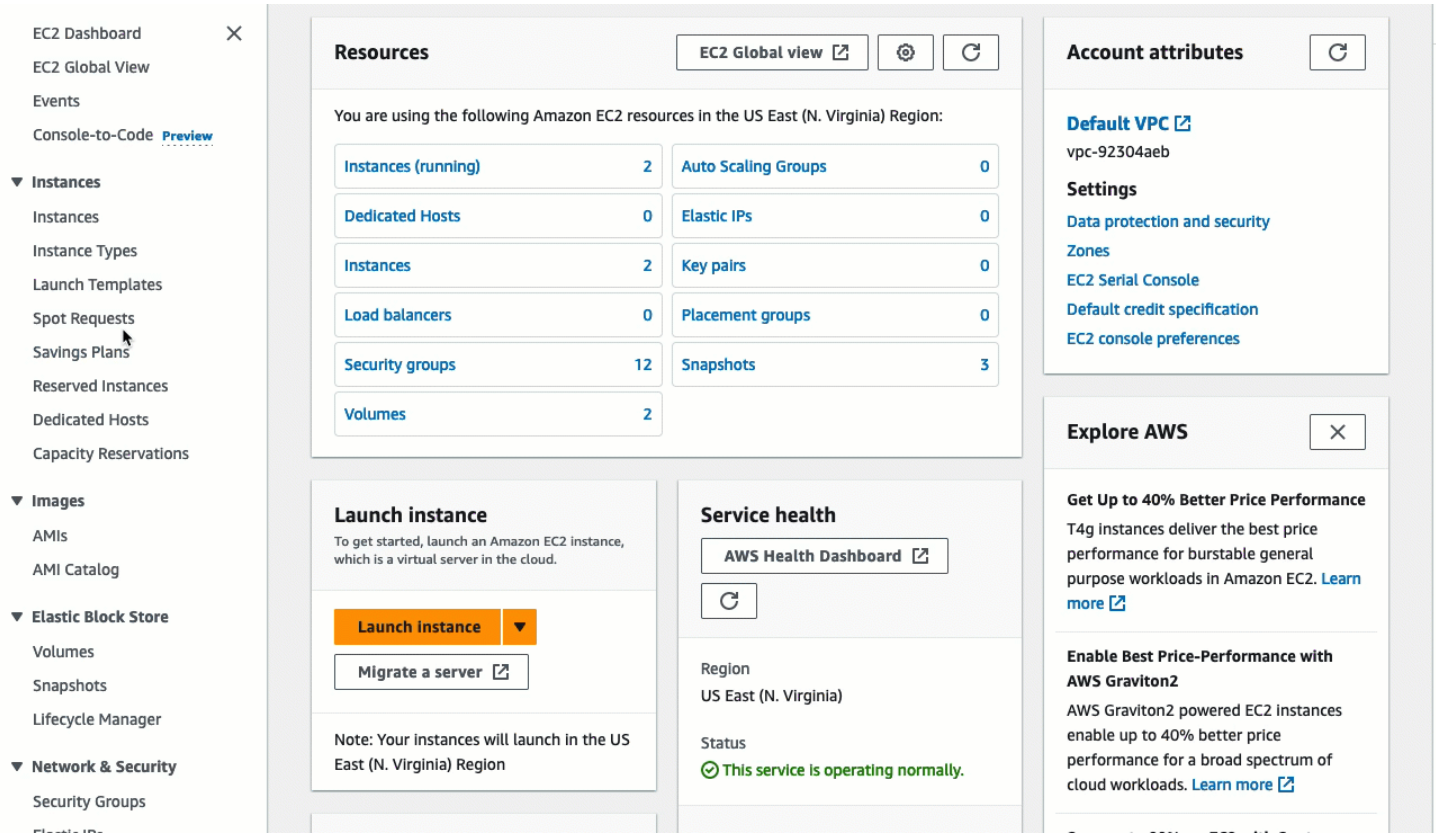
      Amazon EC2 memberi Anda saran misalnya keluarga yang sesuai dengan persyaratan yang Anda tentukan.

3. Untuk melihat detail setiap jenis instans dalam keluarga instans yang disarankan, pilih Lihat detail keluarga instans yang direkomendasikan.

4. Pilih jenis instans yang memenuhi persyaratan Anda, lalu pilih Tindakan, Lun curkan instans atau Tind akan, Buat template peluncuran.

   Atau, jika Anda memulai proses di wizard instance peluncuran atau halaman template peluncuran, dan Anda lebih suka kembali ke alur asli, catat jenis instans yang ingin Anda gunakan. Kemudian, di wizard instance peluncuran atau template peluncuran, untuk Jenis instans, pilih jenis instans, dan selesaikan prosedur untuk meluncurkan instance atau membuat template peluncuran.

Melihat animasi: Dapatkan saran jenis instans menggunakan pencari tipe instans EC2




# Dapatkan rekomendasi instans EC2 dari Compute Optimizer

AWS Compute Optimizer memberikan rekomendasi Amazon EC2 untuk membantu Anda meningkatkan kinerja, menghemat uang, atau keduanya. Anda dapat menggunakan rekomendasi ini untuk memutuskan apakah akan mengubah ke jenis instance baru.

Untuk membuat rekomendasi, Compute Optimizer menganalisis spesifikasi instans dan metrik pemanfaatan yang ada. Data yang dikompilasi kemudian digunakan untuk merekomendasikan tipe instans Amazon EC2 yang paling mampu menangani beban kerja yang sudah ada. Rekomendasi ditampilkan bersama dengan harga instans per jam. Untuk informasi selengkapnya, lihat metrik instans Amazon EC2 di Panduan AWS Compute Optimizer Pengguna.

Daftar Isi


- Persyaratan
- Klasifikasi temuan
- Melihat rekomendasi
- Pertimbangan untuk mengevaluasi rekomendasi

## Persyaratan

Untuk mendapatkan rekomendasi dari Compute Optimizer, Anda harus terlebih dahulu memilih Compute Optimizer. Untuk informasi selengkapnya, lihat Memulai dengan AWS Compute Optimizer dalam Panduan Pengguna AWS Compute Optimizer .

Compute Optimizer menghasilkan rekomendasi untuk beberapa jenis instans, tetapi tidak semua jenis instans. Jika Anda menggunakan jenis instans yang tidak didukung, Compute Optimizer tidak akan menghasilkan rekomendasi. Untuk daftar jenis instans yang didukung, lihat persyaratan instans Amazon EC2 di Panduan AWS Compute Optimizer Pengguna.

## Klasifikasi temuan

Compute Optimizer mengklasifikasikan temuannya untuk instans EC2 sebagai berikut:

- Under-provisionedInstans EC2 dianggap kurang disediakan jika setidaknya satu spesifikasi instans Anda, seperti CPU, memori, atau jaringan, tidak memenuhi persyaratan kinerja beban kerja Anda. Under-provisioned Instans EC2 dapat menyebabkan kinerja aplikasi yang buruk.
- Over-provisionedInstans EC2 dianggap terlalu disediakan ketika setidaknya satu spesifikasi instans Anda, seperti CPU, memori, atau jaringan, dapat dikurangi ukurannya sambil tetap memenuhi persyaratan kinerja beban kerja Anda, dan ketika tidak ada spesifikasi yang kurang disediakan. Over-provisioned Instans EC2 dapat menyebabkan biaya infrastruktur yang tidak perlu.
- Dioptimalkan – Instans EC2 dianggap dioptimalkan jika semua spesifikasi instans Anda, seperti CPU, memori, dan jaringan, memenuhi kebutuhan performa beban kerja Anda, dan instans tidak disediakan secara berlebihan. Instans EC2 yang dioptimalkan menjalankan beban kerja Anda dengan performa dan biaya infrastruktur yang optimal. Untuk instans yang dioptimalkan, Compute Optimizer dapat sewaktu-waktu merekomendasikan tipe instans generasi baru.
- Tidak ada – Tidak ada rekomendasi untuk instans ini. Hal ini mungkin terjadi jika Anda memilih Compute Optimizer selama kurang dari 12 jam, atau ketika instans berjalan kurang dari 30 jam, atau saat tipe instans tidak didukung oleh Compute Optimizer.

## Melihat rekomendasi

Setelah memilih Compute Optimizer, Anda dapat melihat temuan yang dihasilkan Compute Optimizer untuk instans EC2 Anda di konsol Amazon EC2. Anda kemudian dapat mengakses konsol Compute Optimizer untuk melihat rekomendasi. Jika Anda baru-baru ini ikut serta, temuan mungkin tidak tercermin di konsol EC2 hingga 12 jam.

Untuk melihat rekomendasi untuk instance menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih ID instance untuk membuka halaman detail instance.
4. Pada halaman detail instance, di bagian ringkasan atas, AWS Compute Optimizer cari pencarian. Jika ada temuan, kami menampilkan klasifikasi temuan dan tautan untuk melihat detailnya. Jika tidak, kami menampilkan Tidak ada rekomendasi yang tersedia untuk contoh ini.
5. Jika ada temuan, pilih Lihat detail. Ini membuka halaman Rekomendasi untuk instans EC2 di konsol Compute Optimizer. Jenis instance saat ini diberi label Cur rent. Ada juga hingga tiga rekomendasi jenis instans, berlabel Opsi 1, Opsi 2, dan Opsi 3. Halaman ini juga menampilkan data CloudWatch metrik terbaru untuk instance tersebut.

Untuk melihat rekomendasi untuk semua instans di semua Wilayah

Anda dapat melihat rekomendasi untuk semua instans Amazon EC2 di semua Wilayah menggunakan konsol Compute Optimizer. Untuk informasi selengkapnya, lihat Melihat rekomendasi instans EC2 dan Melihat detail instans EC2 di Panduan AWS Compute Optimizer Pengguna.

## Pertimbangan untuk mengevaluasi rekomendasi

Ketika Anda menerima rekomendasi, Anda harus memutuskan apakah akan menindaklanjutinya. Sebelum mengubah tipe instans, pertimbangkan hal berikut:

- Rekomendasi tidak memprakirakan penggunaan Anda. Rekomendasi didasarkan pada penggunaan historis Anda selama periode waktu 14 hari terakhir. Pastikan untuk memilih tipe instans yang diperkirakan memenuhi kebutuhan sumber daya Anda di masa mendatang.
- Fokus pada metrik grafik untuk menentukan apakah penggunaan aktual lebih rendah daripada kapasitas instans. Anda juga dapat melihat data metrik (rata-rata, puncak, persentil) CloudWatch untuk mengevaluasi lebih lanjut rekomendasi instans EC2 Anda. Misalnya, perhatikan cara metrik persentase CPU berubah sepanjang hari dan apakah ada puncak yang perlu diakomodasi. Untuk informasi selengkapnya, lihat Melihat Metrik yang Tersedia di Panduan CloudWatch Pengguna Amazon.
- Compute Optimizer mungkin menyediakan rekomendasi untuk instans performa yang dapat melonjak, yaitu instans T3, T3a, dan T2. Jika Anda secara berkala melonjak di atas garis dasar, pastikan Anda dapat terus melakukannya berdasarkan vCPU tipe instans baru. Untuk informasi selengkapnya, lihat Konsep kunci untuk instans kinerja yang dapat meledak.

- Jika Anda telah membeli Instans Cadangan, Inst On-Demand ans Anda mungkin ditagih sebagai Instans Cadangan. Sebelum Anda mengubah tipe instans saat ini, pertama-tama evaluasi dampaknya terhadap penggunaan dan cakupan Instans Terpesan.
- Pertimbangkan konversi ke instans generasi yang lebih baru, jika memungkinkan.
- Saat bermigrasi ke keluarga instans yang berbeda, pastikan tipe instans saat ini dan tipe instans yang baru kompatibel. Misalnya, dalam hal virtualisasi, arsitektur, atau tipe jaringan. Untuk informasi selengkapnya, lihat Kompatibilitas untuk mengubah tipe instans.
- Terakhir, pertimbangkan penilaian risiko performa yang diberikan untuk setiap rekomendasi. Risiko kinerja menunjukkan jumlah upaya yang mungkin perlu Anda habiskan untuk memvalidasi apakah jenis instans yang direkomendasikan memenuhi persyaratan kinerja beban kerja Anda. Kami juga menyarankan pengujian beban dan performa yang ketat sebelum dan setelah membuat perubahan apa pun.

## Perubahan jenis instans Amazon EC2

Ketika kebutuhan Anda berubah, Anda mungkin menemukan bahwa instans Amazon EC2 Anda terlalu besar atau terlalu kecil untuk beban kerjanya. Jika demikian, Anda dapat mengubah ukuran instans Anda dengan mengubah tipe instans-nya. Misalnya, jika instans `t2.micro` Anda terlalu kecil untuk beban kerjanya, Anda dapat meningkatkan ukurannya dengan mengubahnya ke tipe instans T2 yang lebih besar, seperti `t2.large`. Atau Anda dapat mengubahnya ke tipe instans lain, seperti `m5.large`. Anda mungkin juga harus mengubah dari generasi sebelumnya ke tipe instans generasi terkini untuk memanfaatkan beberapa fitur, seperti dukungan untuk IPv6.

Jika Anda menginginkan rekomendasi tipe instans yang paling mampu menangani beban kerja yang ada, Anda dapat menggunakan AWS Compute Optimizer. Untuk informasi selengkapnya, lihat Dapatkan rekomendasi instans EC2 dari Compute Optimizer.

Saat mengubah tipe instans, Anda akan mulai membayar tarif tipe instans yang baru. Untuk tarif sesuai permintaan dari semua jenis instans, lihat Harga Amazon EC2 On-Demand .

Untuk menambahkan penyimpanan tambahan ke instans Anda tanpa mengubah jenis instans, tambahkan volume EBS ke instans. Untuk informasi selengkapnya, lihat Melamp irkan volume Amazon EBS ke instans di Panduan Pengguna Amazon EBS.

### Instruksi mana yang harus diikuti?

Ada instruksi yang berbeda untuk mengubah tipe instans. Instruksi yang akan digunakan bergantung pada volume root instans, dan apakah tipe instans itu kompatibel dengan konfigurasi instans saat ini.

Untuk informasi tentang bagaimana kompatibilitas ditentukan, lihat Kompatibilitas untuk mengubah tipe instans.

Gunakan tabel berikut untuk menentukan instruksi mana yang harus diikuti.

| Volume akar | Kompatibilitas | Gunakan petunjuk ini |
|---|---|---|
| EBS | Kompatibel | Ubah tipe instans |
| EBS | Tidak kompatibel | Bermigrasi ke jenis instance baru |
| Penyimpanan instans | Tidak berlaku | Bermigrasi ke jenis instance baru |

## Kompatibilitas untuk mengubah tipe instans

Anda dapat mengubah tipe instans hanya jika konfigurasi instans saat ini kompatibel dengan tipe instans yang Anda inginkan. Jika tipe instans yang Anda inginkan tidak kompatibel dengan konfigurasi instans saat ini, Anda harus meluncurkan instans baru dengan konfigurasi yang kompatibel dengan tipe instans tersebut, lalu memigrasikan aplikasi Anda ke instans baru.

Kompatibilitas ditentukan melalui yang berikut:

Tipe virtualisasi

AMI Linux menggunakan salah satu dari dua tipe virtualisasi: paravirtual (PV) atau mesin virtual perangkat keras (HVM). Jika instans diluncurkan dari PV AMI, Anda tidak dapat mengubah tipe instans yang hanya HVM. Untuk informasi selengkapnya, lihat Tipe virtualisasi. Untuk memeriksa tipe virtualisasi instans Anda, periksa nilai Virtualisasi di panel detail layar Instans di konsol Amazon EC2.

Arsitektur

AMI bersifat spesifik untuk arsitektur prosesor, jadi Anda harus memilih tipe instans dengan arsitektur prosesor yang sama dengan tipe instans saat ini. Misalnya:

- Jika tipe instans saat ini memiliki prosesor berdasarkan arsitektur Arm, Anda dibatasi pada tipe instans yang mendukung prosesor berdasarkan arsitektur Arm, seperti C6g dan M6g.
- Tipe instans berikut adalah satu-satunya tipe instans yang mendukung AMI 32-bit: `t2.nano`, `t2.micro`, `t2.small`, `t2.medium`, `c3.large`, `t1.micro`, `m1.small`, `m1.medium`, dan

`c1.medium`. Jika Anda mengubah tipe instans dari instans 32-bit, Anda dibatasi untuk tipe instans ini.

Adaptor jaringan

Jika Anda beralih dari driver untuk satu adaptor jaringan ke yang lain, pengaturan adaptor jaringan diatur ulang saat sistem operasi membuat adaptor baru. Untuk mengonfigurasi ulang pengaturan, Anda mungkin memerlukan akses ke akun lokal dengan izin administrator. Berikut ini adalah contoh perpindahan dari satu adaptor jaringan ke yang lain:

- AWS PV (instans T2) ke Intel 82599 VF (instans M4)
- Intel 82599 VF (sebagian besar instans M4) ke ENA (instans M5)
- ENA (instans M5) ke ENA bandwidth tinggi (instans M5n)

Jaringan yang ditingkatkan

Tipe instans yang mendukung jaringan yang ditingkatkan memerlukan instalasi driver yang diperlukan. Misalnya, Nitro-based instans memerlukan EBS-backed AMI dengan driver Elastic Network Adapter (ENA) diinstal. Untuk mengubah tipe instans yang tidak mendukung peningkatan jaringan menjadi tipe instans yang mendukung peningkatan jaringan, Anda harus menginstal driver ENA atau driver ixgbevf pada instans tersebut, yang sesuai.

> **Note**
>
> Saat Anda mengubah ukuran instans yang mengaktifkan ENA Ekspres diaktifkan, tipe instans baru juga harus mendukung ENA Ekspres. Untuk daftar tipe instans yang mendukung ENA Ekspres, lihat Tipe instans yang didukung untuk ENA Ekspres. Untuk mengubah tipe instans yang mendukung ENA Ekspres ke tipe instans yang tidak mendukungnya, pastikan ENA Ekspres saat ini tidak diaktifkan sebelum Anda mengubah ukuran instans.

NVMe

Volume EBS diekspos sebagai perangkat blok NVMe pada Nitro-based instance. Jika Anda mengubah dari tipe instans yang tidak mendukung NVMe ke tipe instans yang mendukung NVMe, Anda harus menginstal driver NVMe pada instans Anda. Juga, nama perangkat untuk perangkat yang Anda tentukan dalam pemetaan perangkat blok diganti namanya menggunakan nama perangkat NVMe (`/dev/nvme[0-26]n1`).

[Instans Linux] Oleh karena itu, untuk memasang sistem file saat boot menggunakan`/etc/fstab`, Anda harus menggunakan al UUID/Label ih-alih nama perangkat.

Batas Volume

Jumlah maksimum volume Amazon EBS yang dapat dilampirkan ke instans bergantung pada tipe instans dan ukuran instans. Untuk informasi selengkapnya, lihat Batas volume Amazon EBS untuk instans Amazon EC2.

Anda hanya dapat mengubah ke tipe instans atau ukuran instans yang mendukung jumlah volume yang sama atau yang lebih besar daripada yang saat ini dilampirkan ke instans. Jika Anda mengubah ke tipe instans atau ukuran instans yang tidak mendukung jumlah volume yang saat ini dilampirkan, permintaan akan gagal. Misalnya, jika Anda mengubah dari instans `m7i.4xlarge` dengan 32 volume terlampir ke `m6i.4xlarge`, yang mendukung maksimum 27 volume, permintaan akan gagal.

NitroTPM

Jika Anda meluncurkan instance menggunakan AMI dengan NitroTPM diaktifkan dan jenis instans yang mendukung NitroTPM, instance akan diluncurkan dengan NitroTPM diaktifkan. Anda hanya dapat mengubah ke jenis instance yang juga mendukung NitroTPM.

## Mengubah jenis instans untuk instans Amazon EC2 Anda

Gunakan petunjuk berikut untuk mengubah jenis instans Amazon jika jenis EBS-backed instans yang Anda butuhkan kompatibel dengan konfigurasi instans Anda saat ini. Untuk informasi selengkapnya, lihat the section called “Kompatibilitas”.

Pertimbangan

- Anda harus menghentikan instans sebelum Anda dapat mengubah jenis instansnya. Pastikan Anda merencanakan waktu henti saat instans dihentikan. Menghentikan instans dan mengubah tipe instansnya mungkin memerlukan waktu beberapa menit, lalu memulai ulang instans Anda mungkin memerlukan waktu yang bervariasi, tergantung skrip pemulaian aplikasi Anda. Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.
- Saat Anda berhenti dan memulai sebuah instans, kami memindahkan instans tersebut ke perangkat keras baru. Jika instance Anda memiliki alamat IPv4 publik, yang bukan IP Elastic, kami merilis alamat tersebut dan memberikan instans Anda alamat IPv4 publik baru. Untuk informasi selengkapnya tentang perilaku alamat IP sepanjang siklus hidup instance, lihatPerbedaan antara status instance.

- Anda tidak dapat mengubah tipe instans dari Instans Spot.
- [Instans Windows] Kami menyarankan Anda memperbarui paket AWS driver PV sebelum mengubah jenis instans. Untuk informasi selengkapnya, lihat the section called “Mutakhirkan driver PV”.
- Jika instans Anda berada dalam grup Penskalaan Otomatis, layanan Penskalaan Otomatis Amazon EC2 menandai instans yang dihentikan sebagai tidak sehat, dan mungkin menghentikannya dan meluncurkan instance pengganti. Untuk mencegahnya, Anda dapat menangguhkan proses penskalaan untuk grup saat Anda mengubah tipe instans. Untuk informasi selengkapnya, lihat Menangguhkan dan melanjutkan proses grup Auto Scaling dalam Panduan Pengguna Amazon EC2 Auto Scaling.
- Saat Anda mengubah tipe instans dari instans dengan volume penyimpanan instans NVMe, instans yang diperbarui mungkin memiliki volume penyimpanan instans tambahan karena semua volume penyimpanan instans NVMe tersedia meskipun tidak ditentukan dalam pemetaan perangkat blok AMI atau instans. Jika tidak, instans yang diperbarui memiliki jumlah volume penyimpanan instans yang sama dengan yang Anda tentukan saat meluncurkan instans asli.
- Jumlah maksimum volume Amazon EBS yang dapat dilampirkan ke instans bergantung pada tipe instans dan ukuran instans. Anda tidak dapat mengubah ke tipe instans atau ukuran instans yang tidak mendukung jumlah volume yang sudah dilampirkan ke instans Anda. Untuk informasi selengkapnya, lihat Batas volume Amazon EBS untuk instans Amazon EC2.
- [Instans Linux] Anda dapat menggunakan `AWSSupport-MigrateXenToNitroLinux` runbook untuk memigrasikan instans Linux yang kompatibel dari tipe instance Xen ke tipe instance Nitro. Untuk informasi selengkapnya, lihat AWSSupport-MigrateXenToNitroLinux runbook di referensi buku runbook Otomatisasi AWS Systems Manager .
- [Instans Windows] Untuk panduan tambahan tentang migrasi instans Windows yang kompatibel dari tipe instans Xen ke jenis instans Nitro, lihat Migrasi ke jenis instans generasi terbaru.

Untuk mengubah jenis instans dari EBS-backed instans Amazon

1. (Opsional) Jika tipe instans yang baru memerlukan driver yang tidak diinstal pada instans yang ada, Anda harus terhubung ke instans dan menginstal driver terlebih dahulu. Untuk informasi selengkapnya, lihat Kompatibilitas untuk mengubah tipe instans.
2. [Instans Windows] Jika Anda mengonfigurasi instans Windows untuk menggunakan pengalamatan IP statis dan Anda mengubah dari jenis instans yang tidak mendukung jaringan yang disempurnakan ke jenis instans yang mendukung jaringan yang disempurnakan, Anda mungkin mendapatkan peringatan tentang potensi konflik alamat IP saat mengonfigurasi ulang

pengalamatan IP statis. Untuk mencegahnya, aktifkan DHCP pada antarmuka jaringan untuk instans Anda sebelum Anda mengubah tipe instans. Dari instance Anda, buka Network and Sharing Center, buka Internet Protocol Version 4 (TCP/IPv4) Proper ties untuk antarmuka jaringan, dan pilih Dapatkan alamat IP secara otomatis. Ubah tipe instans dan konfigurasikan kembali pengalamatan IP statis pada antarmuka jaringan.

3. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
4. Di panel navigasi, pilih Instans.
5. Pilih instans dan pilih Status instans, Hentikan instans. Ketika diminta konfirmasi, pilih Berhenti. Hal ini dapat memerlukan waktu beberapa menit sampai instans berhenti.
6. Dengan instans yang masih dipilih, klik Tindakan, Pengaturan instans, Ubah tipe instans. Opsi ini berwarna abu-abu jika status instans tidak `stopped`.
7. Pada halaman Ubah tipe instans, lakukan hal berikut:

   a. Untuk Tipe instans, pilih tipe instans yang Anda inginkan.

      Jika tipe instans tidak ada dalam daftar, maka instans itu tidak kompatibel dengan konfigurasi instans Anda. Sebagai gantinya, gunakan instruksi berikut: Bermigrasi ke jenis instans baru dengan meluncurkan instans EC2 baru.

   b. (Opsional) Jika jenis instans yang Anda pilih mendukung pengoptimalan EBS, pilih EBS-optimized untuk mengaktifkan pengoptimalan EBS, atau batalkan pilihan EBS-optimized untuk menonaktifkan pengoptimalan EBS.

      Jika jenis instans yang Anda pilih adalah EBS yang dioptimalkan secara default, EBS-optimized dipilih dan Anda tidak dapat membatalkan pilihannya.

   c. (Opsional) Konfigurasikan opsi vCPU pada jenis instance baru.

      Saat Anda mengubah jenis instans instans yang sudah ada, Amazon EC2 menerapkan pengaturan opsi CPU dari instans yang ada ke instans baru, jika memungkinkan. Jika jenis instance baru tidak mendukung pengaturan tersebut, opsi CPU diatur ulang ke N one. Opsi ini menggunakan jumlah default vCPU untuk jenis instans baru.

      Jika jenis instans yang Anda pilih mendukung konfigurasi vCPU, pilih Tentukan opsi CPU di panel detail lanjutan untuk mengonfigurasi vCPU untuk jenis instans baru Anda.

   d. Pilih Ubah untuk menerima pengaturan baru.

8. Untuk memulai instans, pilih instans dan pilih Status instans, Mulai instans. Hal ini diperlukan waktu beberapa menit hingga instans memasuki status `running`. Jika instans Anda tidak akan dimulai, lihat Pemecahan masalah dalam mengubah tipe instans.
9. [Instans Windows] Jika instans Anda menjalankan Windows Server 2016 atau Windows Server 2019 dengan EC2Launch v1, sambungkan ke instans Windows Anda dan jalankan PowerShell skrip EC2Launch berikut untuk mengonfigurasi instance setelah jenis instans diubah.

> **Important**
>
> Kata sandi administrator akan diatur ulang ketika Anda mengaktifkan skrip EC2Launch inisialisasi instans. Anda dapat memodifikasi file konfigurasi untuk menonaktifkan pengaturan ulang kata sandi administrator dengan menentukannya di pengaturan untuk tugas inisialisasi. Untuk langkah-langkah tentang cara menonaktifkan pengaturan ulang kata sandi, lihat Meng onfigurasi tugas inisialisasi (EC2Launch) atau Meng ubah pengaturan (EC2launch v2).

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
Schedule
```

## Bermigrasi ke jenis instans baru dengan meluncurkan instans EC2 baru

Anda dapat mengubah jenis instans EC2 hanya jika itu adalah EBS-backed instance dengan konfigurasi yang kompatibel dengan jenis instans baru yang Anda inginkan. Jika tidak, jika konfigurasi atau instans Anda tidak kompatibel dengan jenis instans baru, atau instans berbasis penyimpanan instans, Anda harus meluncurkan instance pengganti yang kompatibel dengan jenis instans yang Anda inginkan. Untuk informasi selengkapnya tentang bagaimana kompatibilitas ditentukan, lihatKompatibilitas untuk mengubah tipe instans.

Ikhtisar proses migrasi

- Cadangkan data pada instance asli.
- Luncurkan instans baru dengan konfigurasi yang kompatibel dengan jenis instans baru yang Anda inginkan, melampirkan volume EBS apa pun yang dilampirkan ke instans asli Anda.
- Instal aplikasi Anda di instance baru Anda.
- Pulihkan data apa pun.

- Jika instance asli memiliki alamat IP Elastic, Anda harus mengaitkannya dengan instans baru untuk memastikan bahwa pengguna dapat terus menggunakan aplikasi Anda tanpa gangguan.

Untuk memigrasikan instance ke instance baru

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Buat cadangan data yang masih Anda butuhkan sebagai berikut:
   - Sambungkan ke instance Anda dan salin data pada volume penyimpanan instans Anda ke penyimpanan persisten.
   - Buat snapshot volume EBS Anda sehingga Anda dapat membuat volume baru dengan data yang sama, atau melepaskan volume dari instance asli sehingga Anda dapat melampirkannya ke instance baru.
3. Di panel navigasi, pilih Instans.
4. Pilih Luncurkan Instans. Saat Anda mengonfigurasi instans, lakukan hal berikut:
   a. Pilih AMI yang mendukung jenis instance yang Anda inginkan. Misalnya, Anda harus memilih AMI yang mendukung jenis prosesor dari jenis instance baru. Juga, jenis instans generasi saat ini memerlukan AMI HVM.
   b. Pilih tipe instans baru yang Anda inginkan. Jika tipe instans yang Anda inginkan tidak tersedia, maka instans itu tidak kompatibel dengan konfigurasi AMI yang Anda pilih.
   c. Jika Anda ingin mengizinkan lalu lintas yang sama untuk mencapai instance baru, pilih VPC dan grup keamanan yang sama yang digunakan dengan instance asli.
   d. Saat Anda selesai mengonfigurasi instans baru, selesaikan langkah-langkah untuk memilih pasangan kuci dan meluncurkan instans Anda. Hal ini diperlukan waktu beberapa menit hingga instans memasuki status `running`.
5. Jika Anda mencadangkan data ke snapshot EBS, buat volume dari snapshot, lalu lamp irkan volume ke instance baru.

   Untuk memindahkan volume EBS dari instance asli ke instans baru, le paskan volume dari instance asli, lalu lamp irkan volume ke instance baru.
6. Instal aplikasi Anda dan perangkat lunak yang diperlukan pada instans baru.
7. Pulihkan data apa pun yang Anda cadangkan dari volume penyimpanan instans dari instans asli.
8. Jika instance asli memiliki alamat IP Elastic, tetapkan ke instance baru sebagai berikut:
   a. Di panel navigasi, pilih IP Elastis.

b. Pilih alamat IP Elastis yang terkait dengan instans asli dan pilih Tindakan, Pisahkan ke Elastis. Saat diminta konfirmasi, pilih Ya, Nonaktifkan.

c. Dengan alamat IP Elastis masih dipilih, pilih Tndakan, Kaitkan alamat IP Elastis.

d. Untuk tipe Resource, pilih instans.

e. Misalnya, pilih instance baru.

f. (Opsional) Untuk Alamat IP privat, tentukan alamat IP privat yang akan dikaitkan dengan alamat IP Elastis.

g. Pilih Kaitkan.

9. (Opsional) Anda dapat mengakhiri instans asli jika sudah tidak diperlukan lagi. Pilih instans, verifikasi bahwa Anda akan menghentikan instans asli dan bukan instans baru (misalnya, periksa nama atau waktu peluncuran), lalu pilih Status instans, Hentikan instans.

## Pemecahan masalah dalam mengubah tipe instans

Gunakan informasi berikut untuk membantu mendiagnosis dan memperbaiki masalah yang mungkin Anda temui saat mengubah tipe instans.

Instans tidak akan dimulai setelah mengubah tipe instans

Kemungkinan penyebab: Persyaratan untuk tipe instans baru tidak terpenuhi

Jika instans Anda tidak bisa boot, kemungkinan salah satu persyaratan untuk tipe instans baru tidak terpenuhi. Untuk informasi selengkapnya, lihat Mengapa instans Linux saya tidak bisa boot setelah saya mengubah tipenya?

Kemungkinan penyebab: AMI tidak mendukung tipe instans

Jika Anda menggunakan konsol EC2 untuk mengubah tipe instans, hanya tipe instans yang didukung oleh AMI yang dipilih yang akan tersedia. Namun, jika Anda menggunakan AWS CLI untuk meluncurkan instance, Anda dapat menentukan AMI dan jenis instans yang tidak kompatibel. Jika AMI dan tipe instans tidak kompatibel, instans tidak dapat dimulai. Untuk informasi selengkapnya, lihat Kompatibilitas untuk mengubah tipe instans.

Kemungkinan penyebab: Instans dalam grup penempatan klaster

Jika instans Anda berada dalam grup penempatan klaster dan, setelah mengubah tipe instans, instans gagal dimulai, coba yang berikut ini:

1. Hentikan semua instans dalam grup penempatan klaster.

2. Mengubah tipe instans yang terpengaruh.
3. Mulai semua instans dalam grup penempatan klaster.

Aplikasi atau situs web tidak dapat dijangkau dari internet setelah mengubah tipe instans

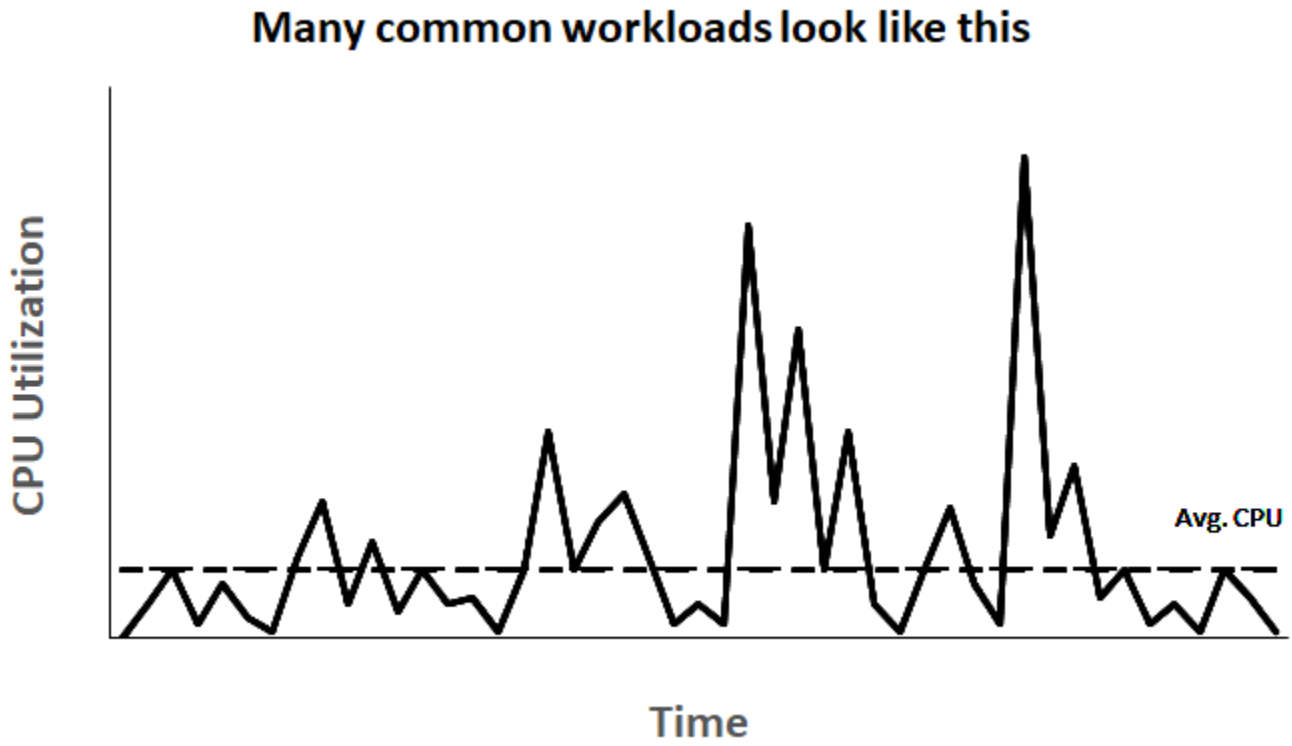
Kemungkinan penyebabnya: Alamat IPv4 publik dirilis

Saat mengubah tipe instans, Anda harus menghentikan instans tersebut terlebih dahulu. Saat Anda menghentikan instans, kami merilis alamat IPv4 publik dan memberi instans Anda alamat IPv4 publik baru.

Untuk mempertahankan alamat IPv4 publik antara penghentian instans dan mulai, sebaiknya gunakan alamat IP Elastic. Untuk informasi selengkapnya, lihat Alamat Elastic IP.

## Instance performa yang dapat melonjak

Banyak beban kerja tujuan umum secara rata-rata tidak sibuk, dan tidak memerlukan performa CPU berkelanjutan tingkat tinggi. Grafik berikut menggambarkan pemanfaatan CPU untuk banyak beban kerja umum yang dijalankan pelanggan di AWS Cloud saat ini.



Beban kerja pemanfatan CPU rendah hingga sedang ini menyebabkan pemborosan siklus CPU dan, akibatnya, Anda harus membayar lebih dari yang digunakan. Untuk mengatasi hal ini, Anda dapat memanfaatkan instans tujuan umum yang dapat melonjak berbiaya rendah, yang merupakan instans T.

Keluarga instans T menyediakan kemampuan untuk melonjak di atas garis dasar bagi performa CPU dasar kapan saja selama yang diperlukan. CPU acuan ditetapkan untuk memenuhi kebutuhan sebagian besar beban kerja tujuan umum, termasuk layanan mikro skala besar, server web, basis data kecil dan menengah, logging data, repositori kode, desktop virtual, lingkungan pengembangan dan pengujian, serta aplikasi penting untuk bisnis. Instans T menawarkan keseimbangan sumber daya komputasi, memori, dan jaringan, serta menyediakan cara yang paling hemat biaya untuk menjalankan spektrum aplikasi tujuan umum yang luas, yang memiliki pemanfaatan CPU rendah hingga sedang. Instans tersebut dapat menghemat biaya Anda hingga 15% jika dibandingkan dengan instans M, serta dapat menghasilkan penghematan biaya lebih dengan ukuran instans yang lebih kecil dan lebih ekonomis, menawarkan 2 vCPU dan 0,5 GiB memori. Ukuran instans T yang lebih kecil, seperti nano, mikro, kecil, dan menengah, sangat cocok untuk beban kerja yang membutuhkan sejumlah kecil memori dan tidak mengharapkan penggunaan CPU yang tinggi.

> **Note**
>
> Topik ini menjelaskan CPU yang dapat melonjak. Untuk informasi tentang performa jaringan yang dapat melonjak, lihat Bandwidth jaringan instans Amazon EC2.

## Tipe instans yang dapat melonjak EC2

Instans burstable EC2 terdiri dari tipe instans T4g, T3a, dan T3, dan tipe instans T2 generasi sebelumnya.

Tipe instans T4g adalah instans yang dapat melonjak generasi terbaru. Tipe instans ini memberikan harga terbaik untuk performa, dan menyediakan biaya semua tipe instans EC2 yang terendah. Jenis instance T4g didukung oleh prosesor Arm-based AWS Graviton2 dengan dukungan ekosistem yang luas dari vendor sistem operasi, vendor perangkat lunak independen, dan AWS layanan dan aplikasi populer.

Tabel berikut merangkum perbedaan utama antara tipe-tipe instans yang dapat melonjak.

| Tipe | Deskripsi | Keluarga prosesor |
|---|---|---|
| Generasi terbaru | | |
| T4g | Jenis instans EC2 berbiaya terendah dengan biaya hingga 40% lebih tinggi price/per formance dan 20% lebih rendah dibandingkan T3 | AWS Prosesor Graviton2 dengan inti Arm Neoverse N1 |
| T3a | Instans berbasis x86 paling murah dengan biaya 10% lebih rendah vs. instans T3 | Prosesor EPYC generasi ke-1 dari AMD |
| T3 | Puncak terbaik price/per formance untuk beban kerja x86 dengan hingga 30% lebih rendah dibandingkan price/per formance instans T2 generasi sebelumnya | Intel Xeon Scalable (prosesor Skylake, Cascade Lake) |
| Generasi sebelumnya | | |
| T2 | Instans yang dapat melonjak generasi sebelumnya | Prosesor Intel Xeon |

Untuk informasi tentang harga instans dan spesifikasi tambahan, lihat Harga Amazon EC2 dan Tipe Instans Amazon EC2. Untuk informasi tentang performa jaringan yang dapat melonjak, lihat Bandwidth jaringan instans Amazon EC2.

Jika Anda membuat instans Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan `t2.micro` instans secara gratis (atau `t3.micro` instans di Wilayah yang `t2.micro` tidak tersedia) dalam batas penggunaan tertentu. Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan`t3.micro`,, `t3.small`t4g.micro, jenis `t4g.small` instans selama 6 bulan atau hingga kredit Anda habis. Untuk informasi selengkapnya, lihat AWS Tingkat Gratis.

Opsi pembelian yang didukung untuk instans T

- On-Demand Contoh
- Instans Terpesan
- Instans Khusus (khusus T3)
- Host Khusus (khusus T3, hanya dalam mode `standard`)
- Instans Spot

Untuk informasi selengkapnya, lihat Opsi penagihan dan pembelian Amazon EC2.

Daftar Isi


- Praktik terbaik
- Konsep kunci untuk instans kinerja yang dapat meledak
- Mode tidak terbatas untuk instans performa yang dapat melonjak
- Mode standar untuk instans performa yang dapat melonjak
- Konfigurasikan instans kinerja yang dapat meledak
- Memantau kredit CPU untuk instans yang dapat meledak


## Praktik terbaik

Ikuti praktik terbaik ini untuk mendapatkan keuntungan maksimal dari instans performa yang dapat melonjak.

- Pastikan ukuran instans yang Anda pilih memenuhi persyaratan memori minimum sistem operasi dan aplikasi Anda. Sistem operasi dengan antarmuka pengguna grafis yang menggunakan banyak memori dan sumber daya CPU (misalnya, Windows) mungkin memerlukan ukuran instans `t3.micro` atau yang lebih besar untuk banyak kasus penggunaan. Seiring bertambahnya kebutuhan memori dan CPU untuk beban kerja Anda dari waktu ke waktu, Anda memiliki fleksibilitas dengan instans T untuk menskalakan ke ukuran instans yang lebih besar dengan tipe instans yang sama, atau untuk memilih tipe instans lainnya.
- Aktifkan AWS Compute Optimizer untuk akun Anda dan tinjau rekomendasi Compute Optimizer untuk beban kerja Anda. Compute Optimizer dapat membantu menilai apakah ukuran instans harus ditingkatkan untuk meningkatkan performa atau diperkecil untuk penghematan biaya. Compute Optimizer juga dapat merekomendasikan tipe instans yang berbeda berdasarkan skenario Anda. Untuk informasi selengkapnya, lihat Melihat rekomendasi instans EC2 dalam Panduan Pengguna AWS Compute Optimizer .

## Konsep kunci untuk instans kinerja yang dapat meledak

Tipe instans Amazon EC2 tradisional menyediakan sumber daya CPU tetap, sementara instans performa yang dapat melonjak menyediakan tingkat pemanfaatan CPU dasar dengan kemampuan untuk melonjakkan pemanfaatan CPU di atas tingkat dasar. Hal ini memastikan Anda membayar hanya untuk CPU dasar dan lonjakan penggunaan CPU tambahan, sehingga biaya komputasi menjadi lebih rendah. Pemanfaatan dasar dan kemampuan untuk melonjak diatur oleh kredit CPU. Instans performa yang dapat melonjak adalah satu-satunya tipe instans yang menggunakan kredit untuk penggunaan CPU.

Setiap instans performa yang dapat melonjak terus-menerus mendapatkan kredit ketika tetap di bawah garis dasar CPU, dan terus-menerus menghabiskan kredit ketika melonjak di atas garis dasar. Jumlah kredit yang diperoleh atau dihabiskan tergantung pemanfaatan CPU dari instans:

- Jika pemanfaatan CPU di bawah garis dasar, maka kredit yang diperoleh lebih besar dari kredit yang dihabiskan.
- Jika pemanfaatan CPU sama dengan garis dasar, maka kredit yang diperoleh sama dengan kredit yang dihabiskan.
- Jika pemanfaatan CPU lebih tinggi dari garis dasar, maka kredit yang dihabiskan lebih tinggi dari kredit yang diperoleh.

Ketika kredit yang didapatkan lebih besar dari kredit yang dihabiskan, maka perbedaannya disebut kredit yang masih harus diperoleh, yang dapat digunakan kemudian untuk melonjak di atas pemanfaatan CPU dasar. Demikian pula, ketika kredit yang dihabiskan lebih dari kredit yang diperoleh, maka perilaku instans bergantung pada mode konfigurasi kredit—mode Standar atau mode Tak Terbatas.

Dalam mode Standar, ketika kredit yang dihabiskan lebih dari kredit yang didapatkan, maka instans akan menggunakan kredit yang masih harus diperoleh untuk melonjak di atas pemanfaatan CPU dasar. Jika kredit masih harus diperoleh sudah tidak tersisa, maka instans secara bertahap turun ke pemanfaatan CPU dasar dan tidak dapat melonjak di atas dasar sampai instans memperoleh kredit lebih.

Dalam mode Tidak Terbatas, jika instans melonjak di atas pemanfaatan CPU dasar, maka instans menggunakan kredit yang masih harus diperoleh terlebih dahulu untuk melonjak. Jika kredit yang masih harus diperoleh sudah tidak tersisa, maka instans menghabiskan kredit surplus untuk melonjak. Ketika pemanfaatan CPU-nya turun di bawah garis dasar, instans tersebut menggunakan kredit CPU yang didapatkan untuk membayar kredit surplus yang dihabiskan sebelumnya.

Kemampuan untuk mendapatkan kredit CPU untuk mengurangi kredit surplus memungkinkan Amazon EC2 untuk meratakan penggunaan CPU dari sebuah instans selama periode 24 jam. Jika penggunaan CPU rata-rata selama periode 24 jam melebihi baseline, instans akan ditagih untuk penggunaan tambahan dengan tarif tambahan tetap tarif per vCPU-hour.

Daftar Isi


- Konsep utama dan definisi
- Mendapatkan kredit CPU
- Tingkat pendapatan kredit CPU
- Batas akrual kredit CPU
- Masa pakai kredit CPU yang masih harus diperoleh
- Pemanfaatan acuan


## Konsep utama dan definisi

Konsep utama dan definisi berikut yang berlaku untuk instans performa yang dapat melonjak.

Pemanfaatan CPU

> Pemanfaatan CPU adalah persentase unit komputasi EC2 yang dialokasikan yang saat ini digunakan pada instans. Metrik ini mengukur persentase siklus CPU yang dialokasikan yang digunakan pada sebuah instance. CloudWatch Metrik Pemanfaatan CPU menunjukkan penggunaan CPU per instance dan bukan penggunaan CPU per inti. Spesifikasi CPU dasar dari sebuah instans juga didasarkan pada penggunaan CPU per instans. Untuk mengukur pemanfaatan CPU menggunakan Konsol Manajemen AWS atau AWS CLI, lihatMendapatkan statistik untuk instans tertentu.

Kredit CPU

> Satuan dari vCPU-time.
>
> Contoh:
>
> 1 kredit CPU = 1 vCPU * 100% pemanfaatan * 1 menit.
>
> 1 kredit CPU = 1 vCPU * 50% pemanfaatan * 2 menit
>
> 1 kredit CPU = 2 vCPU * 25% pemanfaatan * 2 menit

Pemanfaatan dasar

Pemanfaatan dasar adalah tingkat di mana CPU dapat digunakan untuk saldo kredit bersih nol, ketika jumlah kredit CPU yang diperoleh sesuai dengan jumlah kredit CPU yang digunakan. Pemanfaatan dasar juga dikenal sebagai garis dasar. Pemanfaatan dasar dinyatakan sebagai persentase pemanfaatan vCPU, yang dihitung sebagai berikut: Pemanfaatan dasar% = ( earned/ number jumlah kredit vCPU) /60 menit.

Untuk pemanfaatan dasar setiap tipe instans performa yang dapat melonjak, lihat tabel kredit.

Kredit yang diperoleh

Kredit yang diperoleh secara terus-menerus oleh sebuah instans saat sedang berjalan.

Jumlah kredit yang diperoleh per jam = % pemanfaatan dasar * jumlah vCPU * 60 menit

Contoh

t3.nano dengan 2 vCPU dan 5% pemanfaatan dasar memperoleh 6 kredit per jam, dihitung sebagai berikut:

2 vCPU * 5% pemanfaatan dasar * 60 menit = 6 kredit per jam

Kredit yang dihabiskan atau digunakan

Kredit digunakan secara terus-menerus oleh sebuah instans ketika sedang berjalan.

Kredit CPU yang dihabiskan per menit = Jumlah vCPU * pemanfaatan CPU * 1 menit

Kredit yang masih harus diperoleh

Kredit CPU yang tidak terpakai ketika sebuah instans menggunakan kredit lebih sedikit daripada yang diperlukan untuk pemanfaatan dasar. Dengan kata lain, kredit yang masih harus diperoleh = (Kredit yang didapatkan – Kredit yang digunakan) di bawah pemanfaatan dasar.

Contoh

Jika t3.nano berjalan pada 2% pemanfaatan CPU, yang berada 5% di bawah garis dasar selama satu jam, maka kredit yang masih harus diperoleh dihitung sebagai berikut:

Kredit CPU akumulasi = (Kredit yang diperoleh per jam – Kredit yang digunakan per jam) = 6 – 2 vCPU * 2% pemanfaatan CPU * 60 menit = 6 – 2,4 = 3,6 kredit akumulasi per jam

Batas akrual kredit

Tergantung ukuran instans, tetapi secara umum sama dengan jumlah kredit maksimum yang didapatkan dalam 24 jam.

Contoh

Untuk t3.nano, batas akrual kredit = 24 * 6 = 144 kredit

Kredit yang diluncurkan

Hanya berlaku untuk instans T2 yang dikonfigurasi pada mode Standar. Kredit peluncuran adalah jumlah kredit CPU terbatas yang dialokasikan untuk instans T2 baru sehingga ketika diluncurkan dalam mode Standar, dapat melonjak di atas acuan.

Kredit surplus

Kredit yang dihabiskan oleh sebuah instans setelah menghabiskan saldo kredit yang masih harus diperoleh. Kredit surplus didesain untuk instans yang dapat melonjak agar dapat mempertahankan performa tinggi dalam jangka waktu yang lama, dan hanya digunakan dalam mode Tidak Terbatas. Saldo kredit surplus digunakan untuk menentukan jumlah banyak kredit yang digunakan oleh instans untuk melonjak dalam mode Tidak Terbatas.

Mode standar

Mode konfigurasi kredit yang memungkinkan instans melonjak di atas garis dasar dengan menghabiskan kredit yang telah diperoleh dalam saldo kredit.

Mode tidak terbatas

Mode konfigurasi kredit yang memungkinkan instans untuk melonjak di atas garis dasar dengan mempertahankan pemanfaatan CPU yang tinggi untuk jangka waktu kapan pun diperlukan. Harga instance per jam secara otomatis mencakup semua lonjakan penggunaan CPU jika penggunaan CPU rata-rata dari instance berada pada atau di bawah standar selama periode 24 jam bergulir atau masa pakai instance, mana saja yang lebih pendek. Jika instance berjalan pada pemanfaatan CPU yang lebih tinggi untuk jangka waktu yang lama, instans dapat melakukannya untuk tarif tambahan per vCPU-hour.

Tabel berikut merangkum perbedaan utama kredit antara tipe-tipe instans yang dapat melonjak.

| Tipe | Tipe kredit CPU yang didukung | Mode konfigurasi kredit | Masa pakai kredit CPU akumulasi antara instans mulai dan berhenti |
|---|---|---|---|
| Generasi terbaru | | | |
| T4g | Kredit yang diperoleh, Kredit akrual, Kredit yang digunakan, Surplus kredit (hanya pada mode Tidak Terbatas) | Standar, Tidak Terbatas (default) | 7 hari (kredit bertahan selama 7 hari setelah instans berhenti) |
| T3a | Kredit yang diperoleh, Kredit akrual, Kredit yang digunakan, Surplus kredit (hanya pada mode Tidak Terbatas) | Standar, Tidak Terbatas (default) | 7 hari (kredit bertahan selama 7 hari setelah instans berhenti) |
| T3 | Kredit yang diperoleh, Kredit akrual, Kredit yang digunakan, Surplus kredit (hanya pada mode Tidak Terbatas) | Standar, Tidak Terbatas (default) | 7 hari (kredit bertahan selama 7 hari setelah instans berhenti) |
| Generasi sebelumnya | | | |
| T2 | Kredit yang diperoleh, Kredit akumulasi, Kredit yang digunakan, Kredit peluncuran (mode Standar saja), Kredit surplus (hanya ode Tak Terbatas) | Standar (default), Tidak Terbatas | 0 hari (kredit hilang saat instans berhenti) |

> **Note**
>
> Mode Tidak Terbatas tidak didukung untuk instans T3 yang diluncurkan pada Host Khusus.

### Mendapatkan kredit CPU

Setiap instans performa yang dapat melonjak terus-menerus mendapatkan (pada resolusi tingkat milidetik) tingkat kredit CPU yang ditetapkan per jam, tergantung ukuran instans. Proses penghitungan apakah kredit bertambah atau dihabiskan juga terjadi pada resolusi tingkat milidetik, jadi Anda tidak perlu khawatir tentang pengeluaran kredit CPU yang berlebihan. Lonjakan singkat CPU menggunakan sebagian kecil kredit CPU.

Jika instans performa yang dapat melonjak menggunakan lebih sedikit sumber daya CPU daripada yang diperlukan untuk pemanfaatan dasar (seperti saat menganggur), kredit CPU yang tidak terpakai akan ditambahkan ke saldo kredit CPU. Jika instans performa yang dapat melonjak perlu melonjak di atas tingkat pemanfaatan dasar, instans tersebut menghabiskan kredit yang masih harus diperoleh. Makin banyak kredit yang diperoleh oleh instans performa yang dapat melonjak, makin banyak waktu untuk melonjak melebihi garis dasarnya saat memerlukan lebih banyak pemanfaatan CPU.

Tabel berikut mencantumkan daftar tipe instans performa yang dapat melonjak, tarif kredit CPU yang didapatkan per jam, jumlah maksimum kredit CPU yang diperoleh yang dapat diperoleh sebuah instans, jumlah vCPU per instans, dan pemanfaatan dasar sebagai persentase dari inti penuh (menggunakan satu vCPU).

| Jenis instans | Kredit CPU didapatkan per jam | Kredit maksimum yang dihasilkan yang dapat dikumpulkan* | vCPU*** | Pemanfaatan acuan per vCPU |
|---|---|---|---|---|
| T2 | | | | |
| `t2.nano` | 3 | 72 | 1 | 5% |
| `t2.micro` | 6 | 144 | 1 | 10% |
| `t2.small` | 12 | 288 | 1 | 20% |
| `t2.medium` | 24 | 576 | 2 | 20%** |

| Jenis instans | Kredit CPU didapatkan per jam | Kredit maksimum yang dihasilkan yang dapat dikumpulkan* | vCPU*** | Pemanfaatan acuan per vCPU |
|---|---|---|---|---|
| t2.large | 36 | 864 | 2 | 30%** |
| t2.xlarge | 54 | 1296 | 4 | 22,5% ** |
| t2.2xlarge | 81.6 | 1958.4 | 8 | 17%** |
| T3 | | | | |
| t3.nano | 6 | 144 | 2 | 5%** |
| t3.micro | 12 | 288 | 2 | 10%** |
| t3.small | 24 | 576 | 2 | 20%** |
| t3.medium | 24 | 576 | 2 | 20%** |
| t3.large | 36 | 864 | 2 | 30%** |
| t3.xlarge | 96 | 2304 | 4 | 40%** |
| t3.2xlarge | 192 | 4608 | 8 | 40%** |
| T3a | | | | |
| t3a.nano | 6 | 144 | 2 | 5%** |
| t3a.micro | 12 | 288 | 2 | 10%** |
| t3a.small | 24 | 576 | 2 | 20%** |
| t3a.medium | 24 | 576 | 2 | 20%** |
| t3a.large | 36 | 864 | 2 | 30%** |
| t3a.xlarge | 96 | 2304 | 4 | 40%** |

| Jenis instans | Kredit CPU didapatkan per jam | Kredit maksimum yang dihasilkan yang dapat dikumpulkan* | vCPU*** | Pemanfaatan acuan per vCPU |
|---|---|---|---|---|
| t3a.2xlarge | 192 | 4608 | 8 | 40%** |
| T4g | | | | |
| t4g.nano | 6 | 144 | 2 | 5%** |
| t4g.micro | 12 | 288 | 2 | 10%** |
| t4g.small | 24 | 576 | 2 | 20%** |
| t4g.medium | 24 | 576 | 2 | 20%** |
| t4g.large | 36 | 864 | 2 | 30%** |
| t4g.xlarge | 96 | 2304 | 4 | 40%** |
| t4g.2xlarge | 192 | 4608 | 8 | 40%** |

Catatan

* Jumlah kredit yang dapat diperoleh setara dengan jumlah kredit yang bisa didapatkan dalam periode 24 jam.

** Pemanfaatan acuan persentase dalam tabel adalah per vCPU. Di CloudWatch , Pemakaian CPU ditampilkan per vCPU. Misalnya, pemanfaatan CPU untuk `t3.large` instance yang beroperasi pada tingkat dasar ditampilkan sebagai 30% dalam metrik CloudWatch CPU. Untuk informasi tentang cara menghitung pemanfaatan dasar, lihat Pemanfaatan acuan.

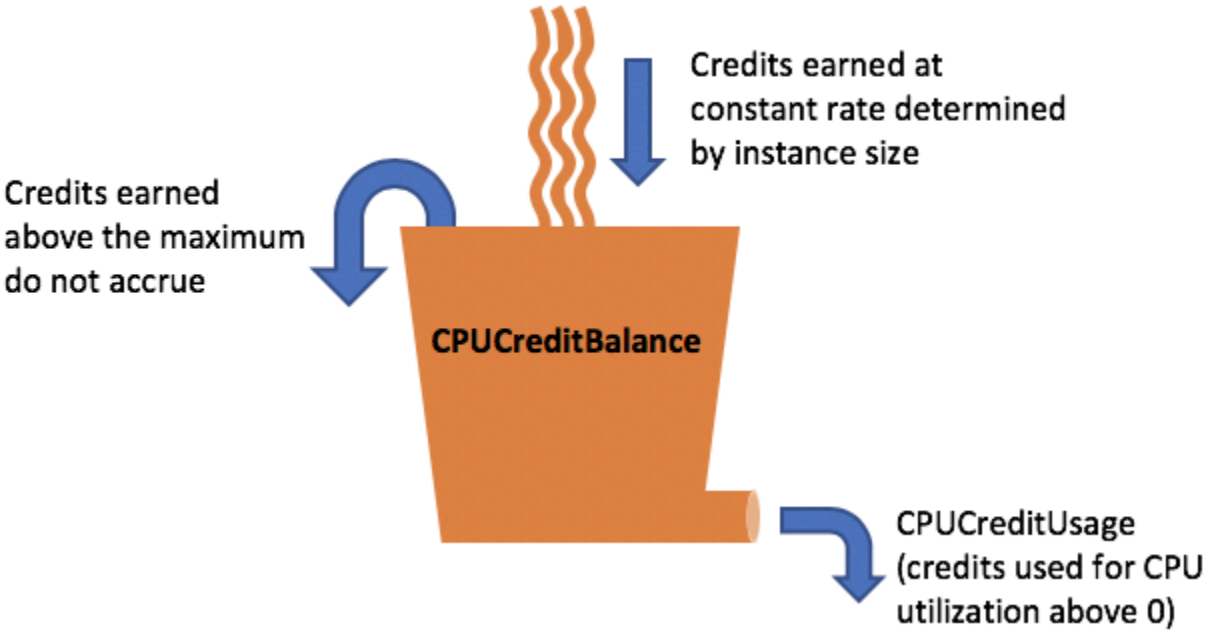
*** Tiap vCPU adalah thread dari inti Intel Xeon atau core AMD EPYC, kecuali untuk instans T2 dan T4g.

Tingkat pendapatan kredit CPU

Jumlah kredit CPU yang didapatkan per jam ditentukan oleh ukuran instans. Misalnya, `t3.nano` mendapatkan enam kredit per jam, sementara `t3.small` mendapatkan 24 kredit per jam. Tabel sebelumnya mencantumkan tingkat pendapatan kredit untuk semua instans.

Batas akrual kredit CPU

Meskipun kredit yang didapatkan tidak pernah kedaluwarsa pada instans yang berjalan, ada batasan jumlah kredit yang didapatkan yang dapat diperoleh sebuah instans. Batas tersebut ditentukan oleh batas saldo kredit CPU. Setelah batas tercapai, semua kredit baru yang didapatkan akan dibuang, seperti yang ditunjukkan pada gambar berikut. Bucket penuh menunjukkan batas saldo kredit CPU, dan spillover menunjukkan pendapatan baru kredit yang melebihi batas.




Batas saldo kredit CPU berbeda untuk setiap ukuran instans. Misalnya, instans `t3.micro` dapat memperoleh maksimum 288 kredit CPU yang didapatkan dalam saldo kredit CPU. Tabel sebelumnya mencantumkan jumlah maksimum kredit yang didapatkan yang dapat diperoleh setiap instans.

Instans T2 Standard juga mendapatkan kredit peluncuran. Kredit peluncuran tidak dihitung dalam batas saldo kredit CPU. Jika instans T2 tidak menghabiskan kredit peluncurannya, dan tetap menganggur selama 24 jam sambil memperoleh kredit yang diperoleh, saldo kredit CPU-nya akan muncul karena melebihi batas. Untuk informasi selengkapnya, lihat Kredit yang diluncurkan.

Instans T4g, T3a, dan T3 tidak mendapatkan kredit peluncuran. Instans ini diluncurkan sebagai `unlimited` secara default, sehingga dapat langsung melonjak saat memulai tanpa kredit peluncuran apa pun. Instans T3 diluncurkan pada peluncuran Host Khusus sebagai `standard` secara default; mode `unlimited` tidak didukung untuk instans T3 pada Host Khusus.

Masa pakai kredit CPU yang masih harus diperoleh

Kredit CPU pada instans yang berjalan tidak kedaluwarsa.

Untuk T2, saldo kredit CPU tidak bertahan antara instans berhenti dan mulai. Jika Anda menghentikan instans T2, instans tersebut kehilangan semua kredit yang masih harus diperoleh.

Untuk T4g, T3a, dan T3, saldo kredit CPU bertahan selama tujuh hari setelah instance berhenti dan kredit hilang setelahnya. Jika Anda memulai instans dalam tujuh hari, tidak ada kredit yang hilang.

Untuk informasi lebih lanjut, lihat`CPUCreditBalance` dalam CloudWatch tabel metrik.

Pemanfaatan acuan

Pem anfaatan dasar adalah tingkat di mana CPU dapat digunakan untuk saldo kredit bersih nol, ketika jumlah kredit CPU yang diperoleh sesuai dengan jumlah kredit CPU yang digunakan. Pemanfaatan dasar juga dikenal sebagai garis dasar.

Pemanfaatan acuan dinyatakan sebagai persentase pemanfaatan vCPU, yang dihitung sebagai berikut:

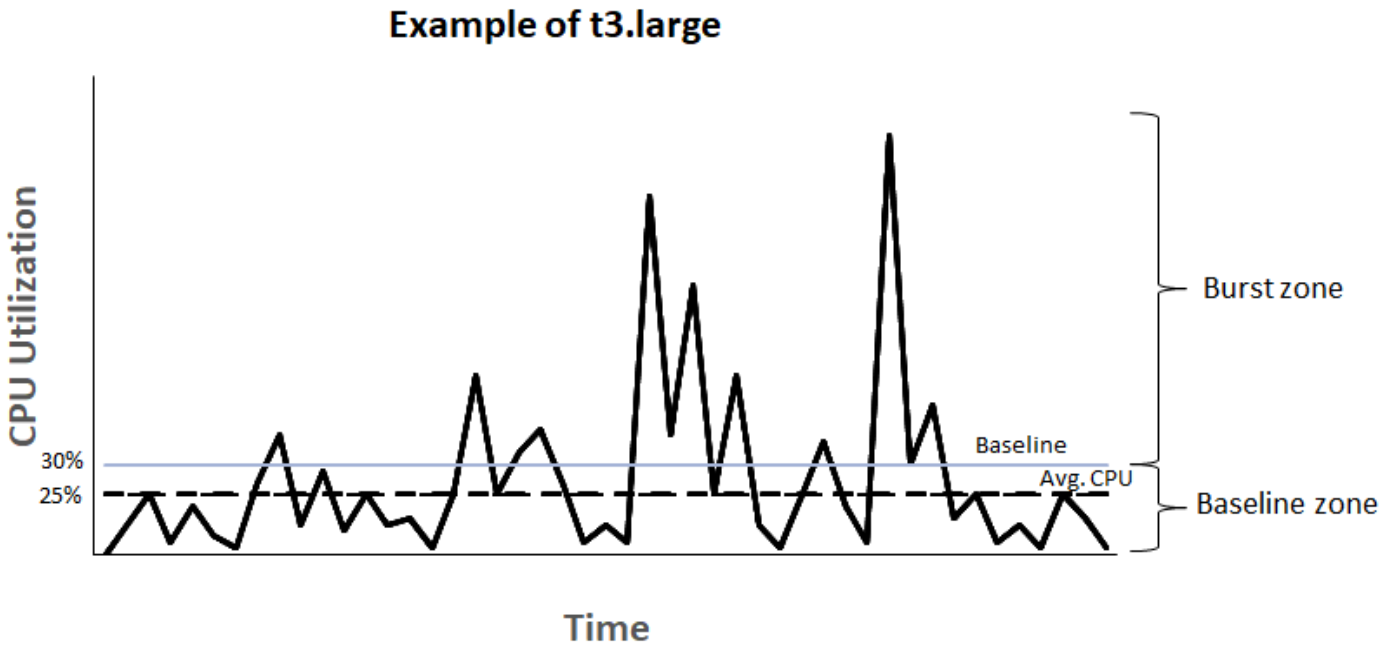
```
(number of credits earned/number of vCPUs)/60 minutes = % baseline
utilization
```

Misalnya, sebuah instans, `t3.nano`dengan 2 vCPU, mendapatkan 6 kredit per jam, menghasilkan pemanfaatan dasar 5%, yang dihitung sebagai berikut:

```
(6 credits earned/2 vCPUs)/60 minutes = 5% baseline utilization
```

Sebuah `t3.large` contoh, dengan 2 vCPU, menghasilkan 36 kredit per jam, menghasilkan pemanfaatan dasar 30% ()`(36/2)/60`.

Grafik berikut memberikan contoh `t3.large` dengan pemanfaatan CPU rata-rata di bawah baseline.




## Mode tidak terbatas untuk instans performa yang dapat melonjak

Instans performa yang dapat melonjak yang dikonfigurasi sebagai `unlimited` dapat mempertahankan pemanfaatan CPU yang tinggi untuk periode waktu kapan pun saat diperlukan. Harga instans per jam secara otomatis mencakup semua fluktuasi penggunaan CPU jika penggunaan CPU rata-rata dari instans sama dengan atau di bawah acuan selama periode 24 jam yang berkelanjutan atau masa pakai instans, mana saja yang lebih pendek.

Untuk sebagian besar beban kerja tujuan umum, instans dikonfigurasi yang sebagai `unlimited` memberikan performa yang cukup tanpa biaya tambahan. Jika instance berjalan pada pemanfaatan CPU yang lebih tinggi untuk waktu yang lama, instans dapat melakukannya dengan tarif tambahan tetap per vCPU-hour. Untuk informasi tentang harga, lihat Harga Amazon EC2 dan Harga Mode Tidak T2/T3/T4 Terbatas Harga Mode

Jika Anda membuat instans Akun AWS sebelum 15 Juli 2025 dan Anda menggunakan `t3.micro` instans `t2.micro` atau di bawah AWS Tingkat Gratis penawaran dan menggunakannya dalam `unlimited` mode, biaya mungkin akan dikenakan jika pemanfaatan rata-rata Anda selama periode 24 jam bergulir melebihi peman fa atan dasar instans.

Instans T4g, T3a, dan T3 diluncurkan secara `unlimited` default (kecuali jika Anda mengubah defaultnya). Jika rata-rata pemanfaatan CPU selama periode 24 jam melebihi garis dasar, Anda dikenai biaya untuk kredit surplus. Jika Anda meluncurkan Instans Spot `unlimited` sebagai dan

berencana segera menggunakannya dan untuk durasi yang singkat, tanpa waktu menganggur untuk memperoleh kredit CPU, Anda dikenai biaya untuk kredit surplus. Kami menyarankan Anda meluncurkan Instans Spot dalam mode standar untuk menghindari pembayaran biaya yang lebih tinggi. Untuk informasi selengkapnya, lihat Kredit surplus dapat dikenakan biaya dan Luncurkan instans kinerja yang meledak.

> **Note**
>
> Instans T3 diluncurkan pada peluncuran Host Khusus sebagai `standard` secara default; mode `unlimited` tidak didukung untuk instans T3 pada Host Khusus.

Daftar Isi


- Konsep mode tak terbatas untuk instans yang dapat meledak
  - Cara kerja instans performa yang dapat melonjak Tidak Terbatas
  - Kapan menggunakan mode tak terbatas versus CPU tetap
  - Kredit surplus dapat dikenakan biaya
  - Berapa biaya kinerja burstable tanpa batas?
  - Tidak ada kredit peluncuran untuk instans T2 Tidak Terbatas
  - Mengaktifkan mode tidak terbatas
  - Yang terjadi pada kredit saat beralih antara Tidak Terbatas dan Standar
  - Memantau penggunaan kredit
- Contoh mode tak terbatas untuk instans yang dapat meledak
  - Contoh 1: Menjelaskan penggunaan kredit dengan T3 Tidak Terbatas
  - Contoh 2: Menjelaskan penggunaan kredit dengan T2 Tidak Terbatas

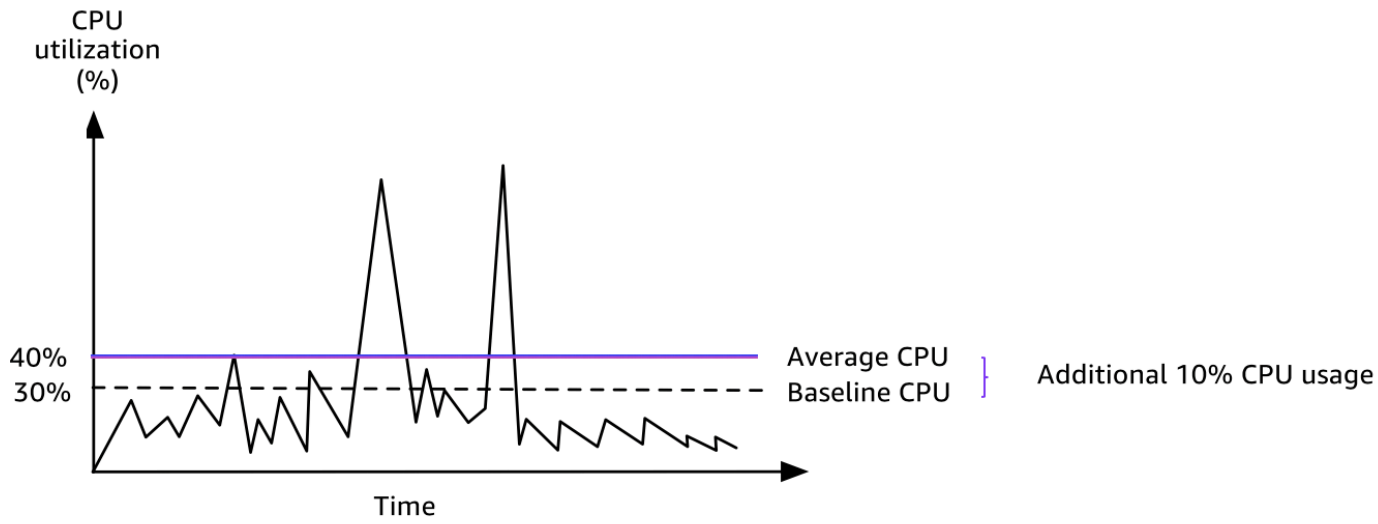

## Konsep mode tak terbatas untuk instans yang dapat meledak

Mode `unlimited` adalah opsi konfigurasi kredit untuk instans performa yang dapat melonjak. Mode ini dapat diaktifkan atau dinonaktifkan kapan saja untuk instans yang berjalan atau dihentikan. Anda dapat menetapkan `unlimited` sebagai opsi kredit default di tingkat akun per Wil AWS ayah, per rangkaian instans kinerja burstable, sehingga semua instans kinerja burstable baru di akun diluncurkan menggunakan opsi kredit default.

Cara kerja instans performa yang dapat melonjak Tidak Terbatas

Jika instans performa yang dapat melonjak yang dikonfigurasi sebagai `unlimited` menghabiskan saldo kredit CPU-nya, Instans tersebut dapat menggunakan kredit surplus untuk melampaui batas dasar. Ketika pemanfaatan CPU-nya turun di bawah garis dasar, instans tersebut menggunakan kredit CPU yang didapatkan untuk membayar kredit surplus yang dihabiskan sebelumnya. Kemampuan untuk mendapatkan kredit CPU untuk mengurangi kredit surplus memungkinkan Amazon EC2 untuk meratakan penggunaan CPU dari sebuah instans selama periode 24 jam. Jika penggunaan CPU rata-rata selama periode 24 jam melebihi baseline, instans akan ditagih untuk penggunaan tambahan dengan tarif tambahan tetap tarif per vCPU-hour.

Grafik berikut menunjukkan penggunaan CPU `t3.large`. Pemanfaatan CPU dasar untuk `t3.large` adalah 30%. Jika instans berjalan pada pemakaian CPU 30% atau kurang secara rata-rata selama periode 24 jam, tidak ada biaya tambahan karena biaya tersebut sudah tercakup oleh harga per jam instans. Namun, jika instance berjalan pada penggunaan CPU rata-rata 40% selama periode 24 jam, seperti yang ditunjukkan pada grafik, instans akan ditagih untuk penggunaan CPU tambahan 10% tambahan dengan tarif tambahan tetap tarif per vCPU-hour.




Untuk informasi selengkapnya tentang pemanfaatan acuan per vCPU untuk setiap tipe instans dan jumlah kredit yang didapatkan oleh setiap tipe instans, lihat tabel kredit.

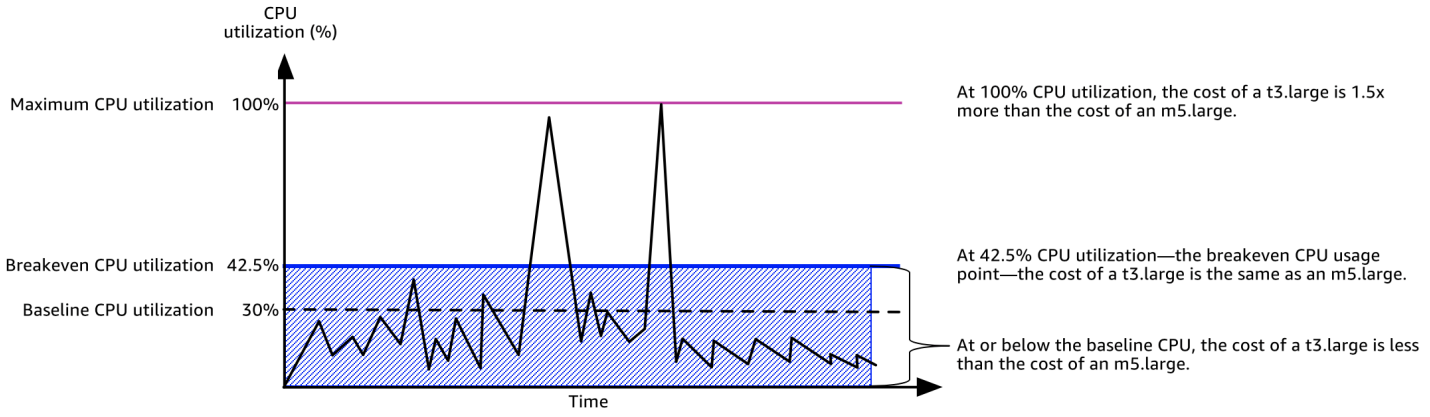
Kapan menggunakan mode tak terbatas versus CPU tetap

Saat menentukan apakah Anda harus menggunakan instans performa yang dapat melonjak dalam mode`unlimited`, seperti T3, atau instans performa tetap, seperti M5, Anda perlu menentukan penggunaan CPU yang impas. Penggunaan CPU yang impas untuk instans performa yang dapat

melonjak adalah titik di mana instans performa yang dapat melonjak harganya sama dengan instans performa tetap. Penggunaan CPU yang impas membantu Anda menentukan hal berikut:

- Jika penggunaan CPU rata-rata selama periode 24 jam sama dengan atau di bawah penggunaan impas CPU, gunakan instans performa yang dapat melonjak di mode `unlimited` sehingga Anda bisa mendapatkan keuntungan dari harga yang lebih rendah dari instans performa yang dapat melonjak sekaligus mendapatkan kinerja yang sama sebagai instans performa tetap.
- Jika penggunaan CPU rata-rata selama periode 24 jam di atas penggunaan impas CPU, instans performa yang dapat melonjak akan lebih mahal daripada instans performa tetap yang berukuran setara. Jika instans T3 terus-menerus melonjak pada 100% CPU, Anda akan membayar sekitar 1,5 kali harga instans M5 yang berukuran setara.

Grafik berikut menunjukkan titik penggunaan impas CPU di mana biaya `t3.large` sama dengan `m5.large`. Titik penggunaan CPU yang impas untuk `t3.large` adalah 42,5%. Jika penggunaan CPU rata-rata adalah 42,5%, biaya menjalankan `t3.large` sama dengan `m5.large`, dan lebih mahal jika penggunaan CPU rata-rata di atas 42,5%. Jika beban kerja membutuhkan penggunaan CPU rata-rata kurang dari 42,5%, Anda dapat memperoleh keuntungan dari harga `t3.large` yang lebih rendah sekaligus mendapatkan performa yang sama sebagai `m5.large`.




Tabel berikut menunjukkan cara menghitung ambang batas penggunaan impas CPU, sehingga Anda dapat menentukan kapan lebih murah untuk menggunakan instans performa yang dapat melonjak dalam mode `unlimited` atau instans performa tetap. Kolom di tabel diberi label A sampai K.

| Jenis instans | vCPU | Harga T3*/ jam | Harga M5*/ jam | Perbedaan harga | Pemanfaatan acuan T3 per vCPU (%) | Biaya per jam vCPU untuk kredit surplus | Biaya per menit vCPU | Menit lonjakan tambahan tersedia per vCPU | % CPU tambahan tersedia | % CPU titik impas |
|---|---|---|---|---|---|---|---|---|---|---|
| A | B | C | D | E = D - C | F | G | H = G / 60 | I = E / H | J = (I / 60) / B | K = F + J |
| t3.large | 2 | $0,0835 | $0,096 | $0,0125 | 30% | $0,05 | $0,00083 | 15 | 12.5% | 42,5% |

| Catatan |
|---|
| * Harga berdasarkan us-east-1 dan Linux OS. |

Tabel tersebut memberikan informasi berikut:

- Kolom A menunjukkan tipe instans, `t3.large`.
- Kolom B menunjukkan jumlah vCPU untuk `t3.large`.
- Kolom C menunjukkan harga `t3.large` per jam.
- Kolom D menunjukkan harga `m5.large` per jam.
- Kolom E menunjukkan perbedaan harga antara `t3.large` dan `m5.large`.
- Kolom F menunjukkan pemanfaatan batas dasar per vCPU dari `t3.large`, yaitu 30%. Pada batas dasar, biaya per jam dari instans mencakup biaya penggunaan CPU.
- Kolom G menunjukkan tarif tambahan tetap per v CPU-hour yang dikenakan biaya instans jika meledak pada CPU 100% setelah kredit yang diperoleh habis.
- Kolom H menunjukkan tarif tambahan tetap per v CPU-minute yang dikenakan biaya instans jika meledak pada CPU 100% setelah kredit yang diperoleh habis.
- Kolom I menunjukkan jumlah menit tambahan yang dapat dilonjakkan oleh `t3.large` per jam pada 100% CPU sementara membayar harga yang sama per jam sebagai `m5.large`.

- Kolom J menunjukkan penggunaan CPU tambahan (dalam %) di atas batas dasar yang instansnya dapat melonjak sementara membayar harga yang sama per jam sebagai `m5.large`.
- Kolom K menunjukkan penggunaan CPU yang impas (dalam %) sehingga `t3.large` dapat melonjak tanpa harus membayar lebih dari `m5.large`. Apa pun di atas ini, dan biaya `t3.large` lebih dari `m5.large`.

Tabel berikut menunjukkan penggunaan CPU yang impas (dalam%) untuk tipe instans T3 dibandingkan dengan tipe instans M5 yang berukuran serupa.

| Tipe instans T3 | Penggunaan CPU yang impas (dalam %) untuk T3 dibandingkan dengan M5 |
|---|---|
| `t3.large` | 42,5% |
| `t3.xlarge` | 52.5% |
| `t3.2xlarge` | 52.5% |

Kredit surplus dapat dikenakan biaya

Jika pemanfaatan CPU rata-rata dari sebuah instans berada pada atau di bawah batas dasar, instans tersebut tidak dikenakan biaya tambahan. Karena sebuah instans memperoleh jumlah kredit maksimum dalam periode 24 jam (misalnya, instans `t3.micro` dapat memperoleh maksimum 288 kredit dalam periode 24 jam), instans tersebut dapat menggunakan kredit surplus hingga maksimum itu tanpa dikenakan biaya.

Namun, jika pemanfaatan CPU tetap di atas batas dasar, instans tidak dapat memperoleh kredit yang cukup untuk membayar surplus kredit yang telah digunakan. Kredit surplus yang tidak dibayar dibebankan dengan tarif tambahan tetap per vCPU-hour. Untuk informasi tentang tarif, lihat Harga Mode Tidak T2/T3/T4g Ter .

Kredit surplus yang digunakan lebih awal dikenai tagihan jika salah satu dari hal berikut terjadi:

- Kredit surplus yang digunakan melebihi jumlah kredit maksimum yang dapat diperoleh instans dalam periode 24 jam. Kredit surplus yang digunakan di atas jumlah maksimum akan ditagihkan pada akhir jam.
- Instans dihentikan atau diakhiri.

- instans dialihkan dari `unlimited` ke `standard`.

Kredit surplus yang dihabiskan dilacak oleh CloudWatch metrik`CPUSurplusCreditBalance`. Kredit surplus yang dibebankan dilacak oleh CloudWatch metrik`CPUSurplusCreditsCharged`. Untuk informasi selengkapnya, lihat CloudWatch Metrik tambahan untuk instans kinerja yang dapat meledak.

Berapa biaya kinerja burstable tanpa batas?

Jika Anda menggunakan kredit surplus dan tidak dibayar dengan kredit yang diperoleh (lihatKredit surplus dapat dikenakan biaya), Anda membayar tarif tambahan tetap per v CPU-hour untuk kredit surplus. Tarif tercantum di T2/T3 batas Harga Mode Tidak Terbatas di  halaman On-Demand Harga Amazon EC2.

Tidak ada kredit peluncuran untuk instans T2 Tidak Terbatas

Instans T2 Standar menerima kredit peluncuran, tetapi tidak dengan instans T2 Tidak Terbatas. Instans T2 Tidak Terbatas dapat melonjak melampaui batas dasar kapan saja tanpa biaya tambahan, selama rata-rata penggunaan CPU berada pada atau di bawah batas dasar selama jangka waktu 24 jam bergulir atau masa pakainya, mana yang lebih pendek. Dengan demikian, instans T2 Tidak Terbatas tidak memerlukan kredit peluncuran untuk mencapai performa tinggi segera setelah peluncuran.

Jika instans T2 dialihkan dari `standard` ke `unlimited`, semua kredit peluncuran yang terkumpul dihapus dari `CPUCreditBalance` sebelum sisa `CPUCreditBalance` diteruskan.

Instans T4g, T3a, dan T3 tidak pernah menerima kredit peluncuran karena diluncurkan dalam mode Unlimited secara default, dan karena itu dapat meledak segera setelah memulai. Konfigurasi kredit mode Unlimited memungkinkan instans T4g, T3a, dan T3 menggunakan CPU sebanyak yang diperlukan untuk melampaui baseline dan selama yang diperlukan.

Mengaktifkan mode tidak terbatas

Anda dapat beralih dari `unlimited` ke `standard`, dan dari `standard` ke `unlimited`, kapan saja pada instans yang berjalan atau dihentikan. Untuk informasi selengkapnya, lihat Konfigurasikan spesifikasi kredit saat peluncuran dan Mengelola spesifikasi kredit instans kinerja yang meledak.

Anda dapat menetapkan `unlimited` sebagai opsi kredit default di tingkat akun per Wil AWS ayah, per rangkaian instans kinerja burstable, sehingga semua instans kinerja burstable baru di

akun diluncurkan menggunakan opsi kredit default. Untuk informasi selengkapnya, lihat Mengelola spesifikasi kredit default untuk akun.

Anda dapat memeriksa apakah instans performa yang dapat melonjak Anda dikonfigurasi sebagai `unlimited` atau `standard` menggunakan konsol Amazon EC2 atau AWS CLI. Untuk informasi selengkapnya, lihat Konfigurasikan instans kinerja yang dapat meledak.

Yang terjadi pada kredit saat beralih antara Tidak Terbatas dan Standar

`CPUCreditBalance`adalah CloudWatch metrik yang melacak jumlah kredit yang diperoleh oleh sebuah instance. `CPUSurplusCreditBalance`adalah CloudWatch metrik yang melacak jumlah kredit surplus yang dihabiskan oleh suatu instance.

Jika Anda mengubah instans yang dikonfigurasi sebagai `unlimited` ke `standard`, hal berikut ini terjadi:

- Nilai `CPUCreditBalance` tetap tidak berubah dan diteruskan.
- Nilai `CPUSurplusCreditBalance` segera dikenakan tagihan.

Jika instans `standard` dialihkan ke `unlimited`, hal berikut ini terjadi:

- Nilai `CPUCreditBalance` yang berisi kredit yang diperoleh yang masih harus dibayar diteruskan.
- Untuk instans T2 Standard, semua kredit peluncuran dihapus dari`CPUCreditBalance` nilai, dan sisanya`CPUCreditBalance` nilai yang mengandung kredit yang diperoleh yang masih harus dibayar diteruskan.

Memantau penggunaan kredit

Untuk melihat apakah instans Anda menghabiskan lebih banyak kredit daripada yang disediakan baseline, Anda dapat menggunakan CloudWatch metrik untuk melacak penggunaan, dan Anda dapat mengatur alarm per jam agar diberi tahu tentang penggunaan kredit. Untuk informasi selengkapnya, lihat Memantau kredit CPU untuk instans yang dapat meledak.

Contoh mode tak terbatas untuk instans yang dapat meledak

Contoh berikut menjelaskan penggunaan kredit untuk instans yang dikonfigurasi sebagai `unlimited`.

Contoh

- Contoh 1: Menjelaskan penggunaan kredit dengan T3 Tidak Terbatas
- Contoh 2: Menjelaskan penggunaan kredit dengan T2 Tidak Terbatas


### Contoh 1: Menjelaskan penggunaan kredit dengan T3 Tidak Terbatas

Dalam contoh ini, Anda melihat pemanfaatan CPU dari instans `t3.nano` yang diluncurkan sebagai `unlimited`, dan caranya menggunakan kredit yang diperoleh dan surplus untuk mempertahankan pemanfaatan CPU.

Instans `t3.nano` memperoleh 144 kredit CPU selama periode 24 jam bergulir, yang dapat ditukarkan dengan 144 menit penggunaan vCPU. Ketika menghabiskan saldo kredit CPU-nya (diwakili oleh CloudWatch metrik`CPUCreditBalance`), ia dapat menghabiskan kelebihan kredit CPU — yang belum diperoleh — untuk meledak selama yang dibutuhkan. Karena instans `t3.nano` memperoleh maksimal 144 kredit dalam jangka waktu 24 jam, instans ini dapat menggunakan kredit surplus hingga maksimum tersebut tanpa langsung dikenakan biaya. Jika menghabiskan lebih dari 144 kredit CPU, instans ini akan dikenakan biaya untuk selisihnya di akhir jam.

Maksud dari contoh tersebut, yang diilustrasikan oleh grafik berikut, adalah untuk menunjukkan bagaimana sebuah instans dapat melonjak menggunakan surplus kredit bahkan setelah instans tersebut menghabiskan `CPUCreditBalance`. Alur kerja berikut mereferensikan titik-titik bernomor pada grafik:

P1 - Pada 0 jam pada grafik, instans diluncurkan sebagai `unlimited` dan langsung mulai mendapatkan kredit. Instans tetap diam sejak diluncurkan, yang artinya pemakaian CPU 0%, sehingga tidak ada kredit yang digunakan. Semua kredit yang tidak terpakai diakumulasi ke dalam saldo kredit. Selama 24 jam pertama, `CPUCreditUsage` berada di 0, dan nilai `CPUCreditBalance` mencapai maksimum 144.

P2 - Untuk 12 jam ke depan, pemanfaatan CPU berada pada 2,5%, yang masih di bawah acuan 5%. Instans mendapatkan lebih banyak kredit daripada yang dibelanjakan, tetapi`CPUCreditBalance` nilai tidak dapat melebihi maksimum 144 kredit.

P3 - Untuk 24 jam ke depan, penggunaan CPU berada pada 7% (di atas acuan), yang membutuhkan penggunaan 57,6 kredit. Instans menggunakan lebih banyak kredit daripada yang diperolehnya, dan nilai `CPUCreditBalance` berkurang menjadi 86,4 kredit.

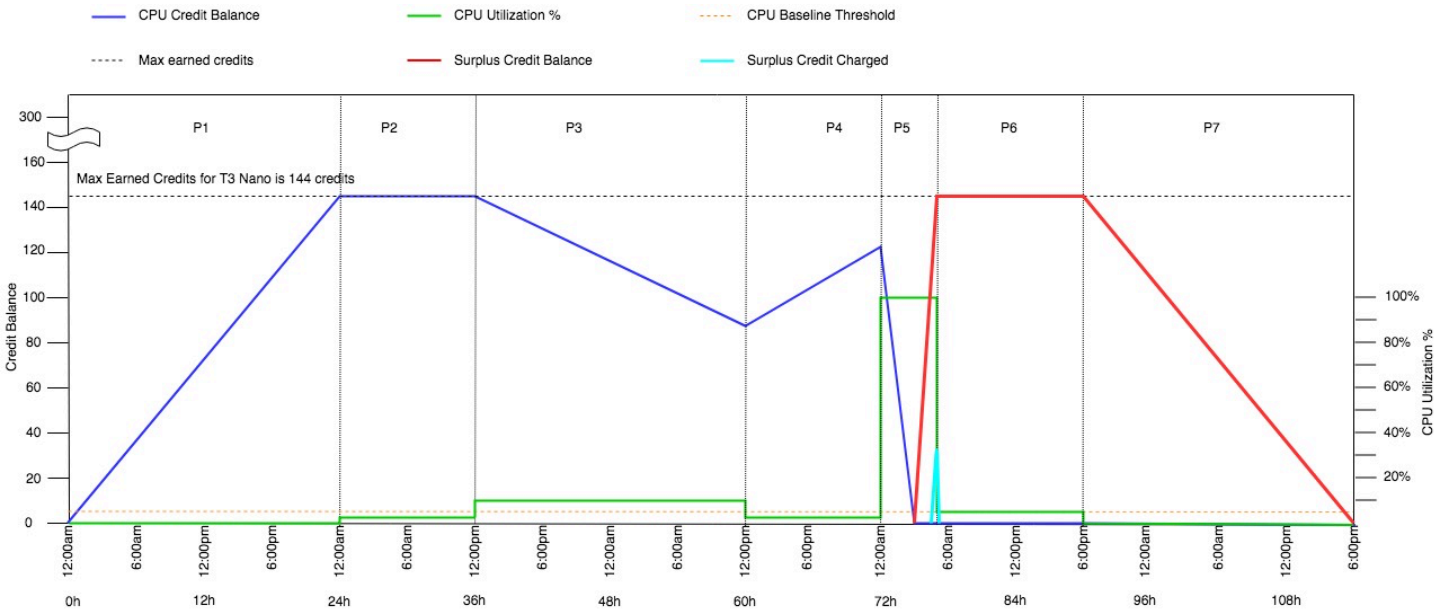
P4 - Selama 12 jam ke depan, pemanfaatan CPU menurun jadi 2,5% (di bawah acuan), yang membutuhkan penggunaan 36 kredit. Pada saat yang sama, instans tersebut mendapatkan 72 kredit.

Instance mendapatkan lebih banyak kredit daripada yang dibelanjakan, dan`CPUCreditBalance` nilai meningkat menjadi 122 kredit.

P5 - Untuk 5 jam ke depan, instans meningkatkan pemanfaatan 100% CPU, dan menggunakan total 570 kredit untuk mempertahankan lonjakan. Sekitar satu jam dalam periode ini, instans menghabiskan seluruh `CPUCreditBalance` sebesar 122 kredit, dan mulai menggunakan kredit surplus untuk mempertahankan penggunaan CPU yang tinggi, dengan total 448 kredit surplus dalam periode ini (570-122=448). Saat nilai `CPUSurplusCreditBalance` mencapai 144 kredit CPU (maksimum yang dapat diperoleh instans `t3.nano` dalam periode 24 jam), kredit surplus yang digunakan setelahnya tidak dapat diimbangi dengan kredit yang diperoleh. Kredit surplus yang dihabiskan setelahnya berjumlah 304 kredit (448-144=304), yang menghasilkan sedikit biaya tambahan pada akhir jam untuk 304 kredit.

P6 - Untuk 13 jam ke depan, pemanfaatan CPU berada pada 5% (batas dasar). Instans tersebut mendapatkan kredit sebanyak yang digunakan, tanpa kelebihan untuk membayar `CPUSurplusCreditBalance`. Nilai `CPUSurplusCreditBalance` tetap sebesar 144 kredit.

P7 - Selama 24 jam terakhir dalam contoh ini, instans tidak aktif dan pemanfaatan CPU adalah 0%. Selama waktu ini, instans memperoleh 144 kredit, yang digunakan untuk membayar `CPUSurplusCreditBalance`.




Contoh 2: Menjelaskan penggunaan kredit dengan T2 Tidak Terbatas

Dalam contoh ini, Anda melihat pemanfaatan CPU dari instans `t2.nano` yang diluncurkan sebagai `unlimited`, dan caranya menggunakan kredit yang diperoleh dan surplus untuk mempertahankan pemanfaatan CPU.

Instans `t2.nano` memperoleh 72 kredit CPU selama periode 24 jam bergulir, yang dapat ditukarkan dengan 72 menit penggunaan vCPU. Ketika menghabiskan saldo kredit CPU-nya (diwakili oleh CloudWatch metrik`CPUCreditBalance`), ia dapat menghabiskan kelebihan kredit CPU — yang belum diperoleh — untuk meledak selama yang dibutuhkan. Karena instans `t2.nano` memperoleh maksimal 72 kredit dalam jangka waktu 24 jam, instans ini dapat menggunakan kredit surplus hingga maksimum tersebut tanpa langsung dikenakan biaya. Jika menghabiskan lebih dari 72 kredit CPU, instans ini akan dikenakan biaya untuk selisihnya di akhir jam.

Maksud dari contoh tersebut, yang diilustrasikan oleh grafik berikut, adalah untuk menunjukkan cara sebuah instans dapat melonjak menggunakan kredit surplus bahkan setelah instans tersebut menghabiskan `CPUCreditBalance`. Anda dapat mengasumsikan bahwa, pada awal lini waktu dalam grafik, instans memiliki saldo kredit yang masih harus didapat dengan besaran yang sama dengan jumlah kredit maksimum yang dapat diperoleh dalam 24 jam. Alur kerja berikut mereferensikan titik-titik bernomor pada grafik:

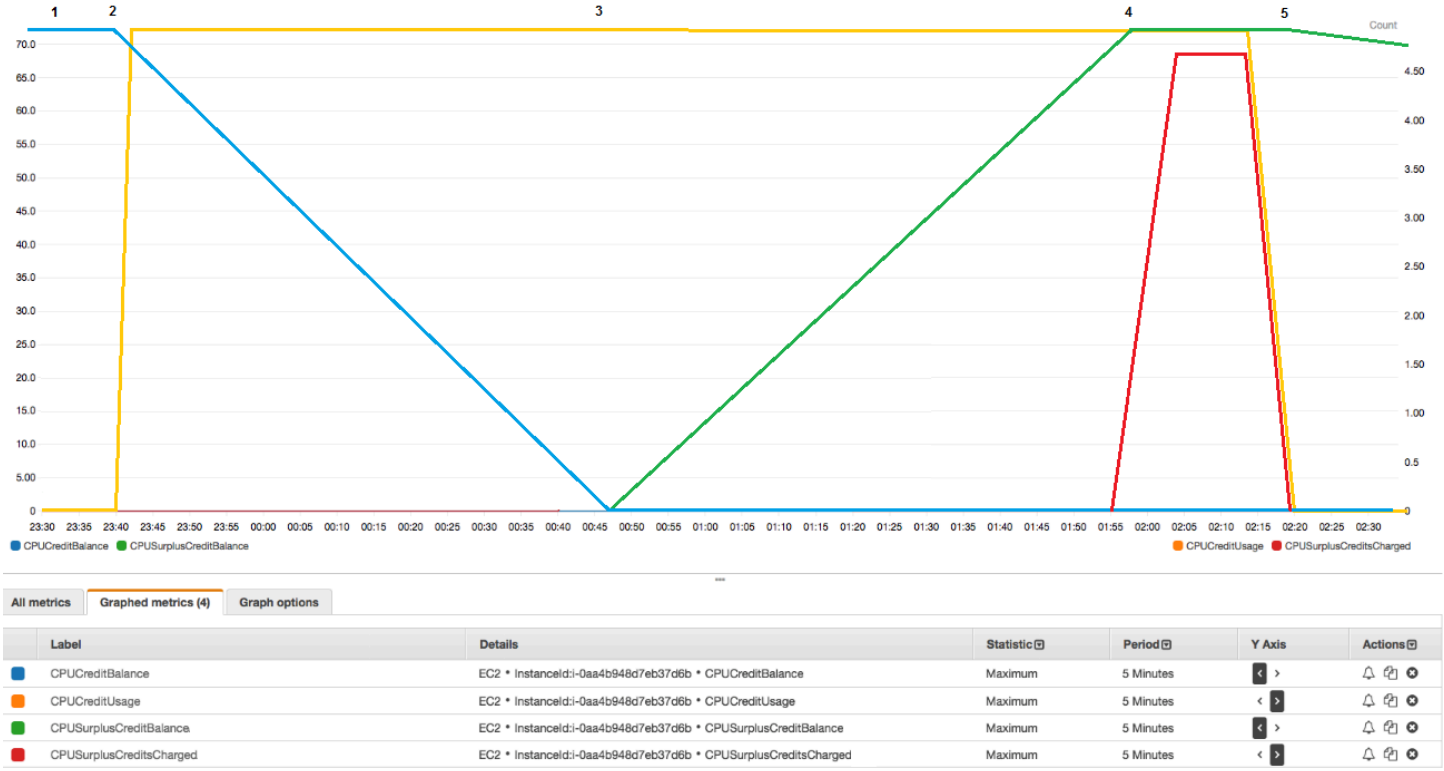
1 – Dalam 10 menit pertama, `CPUCreditUsage` berada di 0, dan nilai `CPUCreditBalance` tetap maksimal sebesar 72.

2 – Pada pukul 23:40, seiring dengan meningkatnya pemanfaatan CPU, instans menggunakan kredit CPU dan nilai `CPUCreditBalance` menurun.

3 – Sekitar pukul 00:47, instans menghabiskan seluruh `CPUCreditBalance`, dan mulai menggunakan kredit surplus untuk mempertahankan pemanfaatan CPU yang tinggi.

4 – Kredit Surplus dihabiskan sampai 01:55, saat nilai `CPUSurplusCreditBalance` mencapai 72 kredit CPU. Jumlah ini sama dengan maksimum yang dapat dihasilkan oleh instans `t2.nano` dalam periode 24 jam. Kredit surplus apa pun yang digunakan setelahnya tidak dapat diimbangi dengan kredit yang diperoleh dalam periode 24 jam, yang menghasilkan sedikit biaya tambahan di akhir jam.

5 – Instans terus menggunakan kredit surplus hingga sekitar pukul 02:20. Pada waktu ini, pemanfaatan CPU berada di bawah batas dasar, dan instans mulai memperoleh kredit sebesar 3 kredit per jam (atau 0,25 kredit setiap 5 menit), yang digunakan untuk membayar `CPUSurplusCreditBalance`. Setelah nilai `CPUSurplusCreditBalance` berkurang hingga menjadi 0, instans mulai mengumpulkan kredit yang diperoleh di `CPUCreditBalance` sebesar 0,25 kredit setiap 5 menit.




Menghitung tagihan (contoh Linux)

Kredit surplus berharga $0,05 per vCPU-hour. Instans menghabiskan sekitar 25 kredit surplus antara 01:55 dan 02:20, yang setara dengan 0,42 v. CPU-hours Biaya tambahan untuk contoh ini adalah 0,42 v CPU-hours x $0. 05/vCPU-jam = $0,021, dibulatkan menjadi $0,02. Berikut adalah tagihan akhir bulan untuk instans T2 Tidak Terbatas ini:

| **Amazon Elastic Compute Cloud running Linux/UNIX** | | |
|---|---|---|
| $0.0058 per On Demand Linux t2.nano Instance Hour | 720.000 Hrs | $4.18 |

| **Amazon Elastic Compute Cloud T2 CPU Credits** | | |
|---|---|---|
| $0.05 per vCPU-Hour of T2 CPU credits | 0.420 vCPU-Hours | $0.02 |

Menghitung tagihan (contoh Windows)

Kredit surplus berharga $0,096 per v. CPU-hour Instans menghabiskan sekitar 25 kredit surplus antara 01:55 dan 02:20, yang setara dengan 0,42 v. CPU-hours Biaya tambahan untuk contoh ini adalah 0,42 v CPU-hours x $0. 096/vCPU-jam = $0,04032, dibulatkan menjadi $0,04. Berikut adalah tagihan akhir bulan untuk instans T2 Tidak Terbatas ini:

| Amazon Elastic Compute Cloud running Windows | | |
|---|---|---|
| $0.0081 per On Demand Windows t2.nano Instance Hour | 720.000 Hrs | $5.83 |

| Amazon Elastic Compute Cloud T2 CPU Credits | | |
|---|---|---|
| $0.096 per vCPU-Hour of T2 CPU credits | 0.420 vCPU-Hours | $0.04 |

Anda dapat mengatur peringatan penagihan agar diberi tahu setiap jam tentang biaya yang timbul, dan mengambil tindakan jika diperlukan.

## Mode standar untuk instans performa yang dapat melonjak

Instans performa yang dapat melonjak yang dikonfigurasi sebagai `standard` cocok untuk beban kerja dengan pemanfaatan CPU rata-rata yang secara konsisten di bawah pemanfaatan CPU batas dasar dari instans. Untuk melonjak di atas batas dasar, instans menggunakan kredit yang telah diakumulasikan dalam saldo kredit CPU. Jika instans hampir kehabisan kredit yang masih harus dibayar, pemanfaatan CPU secara bertahap diturunkan ke batas dasar, sehingga instans tidak mengalami penurunan performa yang tajam saat saldo kredit CPU yang masih harus dibayar habis. Untuk informasi selengkapnya, lihat Konsep kunci untuk instans kinerja yang dapat meledak.

Daftar Isi


- Konsep mode standar untuk instans yang dapat meledak
  - Cara kerja instans performa yang dapat melonjak standar
  - Kredit yang diluncurkan
  - Batas kredit peluncuran
  - Perbedaan antara kredit peluncuran dan kredit yang diperoleh
- Contoh mode standar untuk instans yang dapat meledak
  - Contoh 1: Menjelaskan penggunaan kredit dengan T3 Standa
  - Contoh 2: Menjelaskan penggunaan kredit dengan T2 Standar
    - Periode 1: 1 – 24 jam
    - Periode 2: 25 – 36 jam
    - Periode 3: 37 – 61 jam
    - Periode 4: 62 – 72 jam
    - Periode 5: 73 – 75 jam
    - Periode 6: 76 – 90 jam

- Periode 7: 91 – 96 jam

Konsep mode standar untuk instans yang dapat meledak

Mode `standard` adalah opsi konfigurasi untuk instans performa yang dapat melonjak. Mode ini dapat diaktifkan atau dinonaktifkan kapan saja untuk instans yang berjalan atau dihentikan. Anda dapat menetapkan `standard` sebagai opsi kredit default di tingkat akun per Wil AWS ayah, per rangkaian instans kinerja burstable, sehingga semua instans kinerja burstable baru di akun diluncurkan menggunakan opsi kredit default.

Cara kerja instans performa yang dapat melonjak standar

Saat instans performa yang dapat melonjak dikonfigurasi sebagai `standard` berada dalam status berjalan, instans ini secara terus-menerus memperoleh (pada resolusi tingkat milidetik) set tingkat kredit yang diperoleh per jam. Untuk T2 Standar, saat instans dihentikan, semua kredit yang masih harus dibayar hilang, dan saldo kreditnya direset ke nol. Saat dimulai ulang, instans ini menerima set kredit peluncuran baru, dan mulai mengakumulasi kredit yang diperoleh. Untuk instans Standar T4g, T3a, dan T3, saldo kredit CPU tetap ada selama tujuh hari setelah instans berhenti dan kredit hilang setelahnya. Jika Anda memulai instans dalam tujuh hari, tidak ada kredit yang hilang.

Instans T2 Standar menerima dua jenis kredit CPU: kredit yang diperoleh dan kredit peluncuran. Saat instans T2 Standar berada dalam status berjalan, instans ini secara terus-menerus memperoleh (pada resolusi tingkat milidetik) set tingkat kredit yang diperoleh per jam. Pada awalnya, instans ini belum mendapatkan kredit untuk pengalaman startup yang baik; oleh karena itu, untuk memberikan pengalaman memulai yang baik, instans ini menerima kredit peluncuran di awal, yang digunakan pertama kali saat memperoleh kredit yang diakumulasi.

Instans T4g, T3a, dan T3 tidak menerima kredit peluncuran karena mendukung mode Unlimited. Konfigurasi kredit mode Unlimited memungkinkan instans T4g, T3a, dan T3 menggunakan CPU sebanyak yang diperlukan untuk melampaui baseline dan selama yang diperlukan.

Kredit yang diluncurkan

Instans T2 Standard mendapatkan 30 kredit peluncuran per vCPU saat diluncurkan atau dimulai, dan instans T1 Standard mendapatkan 15 kredit peluncuran. Misalnya, sebuah instans `t2.micro` memiliki satu vCPU dan mendapatkan 30 kredit peluncuran, sementara sebuah instans `t2.xlarge` memiliki empat vCPU dan mendapatkan 120 kredit peluncuran. Kredit peluncuran didesain untuk memberikan pengalaman memulai yang baik untuk memungkinkan instans melonjak segera setelah peluncuran sebelum mereka memperoleh kredit yang diakumulasi.

Kredit peluncuran digunakan terlebih dahulu, sebelum kredit yang diperoleh. Kredit peluncuran yang tidak terpakai diakumulasikan dalam saldo kredit CPU, tetapi tidak dihitung dalam batas saldo kredit CPU. Misalnya, instans `t2.micro` memiliki batas saldo kredit CPU 144 kredit yang diperoleh. Jika instans diluncurkan dan tetap idle selama 24 jam, saldo kredit CPU mencapai 174 (30 kredit peluncuran + 144 kredit yang diperoleh), yang melebihi batas. Namun, setelah instans menggunakan 30 kredit peluncuran, saldo kredit tidak boleh melebihi 144. Untuk informasi selengkapnya tentang batas saldo kredit CPU untuk setiap ukuran instans, lihat tabel kredit.

Tabel berikut mencantumkan alokasi kredit CPU awal yang diterima saat peluncuran atau awal, dan jumlah vCPU.

| Jenis instans | Kredit yang diluncurkan | vCPU |
|---|---|---|
| `t1.micro` | 15 | 1 |
| `t2.nano` | 30 | 1 |
| `t2.micro` | 30 | 1 |
| `t2.small` | 30 | 1 |
| `t2.medium` | 60 | 2 |
| `t2.large` | 60 | 2 |
| `t2.xlarge` | 120 | 4 |
| `t2.2xlarge` | 240 | 8 |

Batas kredit peluncuran

Ada batasan berapa kali instans T2 Standar dapat menerima kredit peluncuran. Batas default-nya adalah 100 peluncuran atau permulaan semua instans T2 Standard yang digabungkan per akun, per Wilayah, per periode 24 jam bergulir. Misalnya, batas tercapai saat satu instans dihentikan dan dimulai 100 kali dalam periode 24 jam, atau saat 100 instans diluncurkan dalam periode 24 jam, atau kombinasi lain yang setara dengan 100 permulaan. Akun baru mungkin memiliki batas bawah, yang akan meningkat seiring waktu berdasarkan penggunaan Anda.

> **Tip**
>
> Untuk memastikan bahwa beban kerja Anda selalu mendapatkan performa yang dibutuhkan, beralihlah ke Mode tidak terbatas untuk instans performa yang dapat melonjak atau pertimbangkan untuk menggunakan ukuran instans yang lebih besar.

Perbedaan antara kredit peluncuran dan kredit yang diperoleh

Tabel berikut mencantumkan perbedaan antara kredit peluncuran dan kredit yang diperoleh.

| | Kredit yang diluncurkan | Kredit yang diperoleh |
|---|---|---|
| Tingkat perolehan kredit | Instans T2 Standar memperoleh 30 kredit peluncuran per vCPU saat diluncurkan atau dimulai.<br><br>Jika instans T2 dialihkan dari `unlimited` ke `standard`, instans ini tidak mendapatkan kredit peluncuran pada saat peralihan. | Setiap instans performa yang dapat melonjak T2 terus-menerus memperoleh (pada resolusi tingkat milidetik) tingkat kredit CPU yang ditetapkan per jam, bergantung pada ukuran instans. Untuk informasi selengkapnya tentang jumlah kredit CPU yang diperoleh per ukuran instans, lihat tabel kredit. |
| Batas perolehan kredit | Batas untuk menerima kredit peluncuran adalah 100 peluncuran atau permulaan semua instans T2 Standard yang digabungkan per akun, per Wilayah, per periode 24 jam bergulir. Akun baru mungkin memiliki batas bawah, yang akan meningkat seiring waktu berdasarkan penggunaan Anda. | Instans T2 tidak dapat mengakumulasi lebih banyak kredit daripada batas saldo kredit CPU. Jika saldo kredit CPU telah mencapai batasnya, kredit apa pun yang diperoleh setelah batas tercapai akan dibuang. Kredit peluncuran tidak termasuk dalam penghitungan batas. Untuk informasi selengkapnya tentang batas saldo kredit CPU untuk tiap ukuran instans T2, lihat tabel kredit. |
| Penggunaan kredit | Kredit peluncuran digunakan terlebih dahulu, sebelum kredit yang diperoleh. | Kredit yang diperoleh hanya digunakan setelah semua kredit peluncuran dihabiskan. |

| | Kredit yang diluncurkan | Kredit yang diperoleh |
|---|---|---|
| Kedaluwarsa kredit | Saat instans T2 Standar berjalan, kredit peluncuran tidak kedaluwarsa. Saat instans T2 Standar berhenti atau dialihkan ke T2 Tidak Terbatas, semua kredit peluncuran hilang. | Saat instans T2 berjalan, kredit yang diperoleh yang diakumulasi tidak kedaluwarsa. Saat instans T2 berhenti, semua kredit yang diperoleh yang diakumulasi akan hilang. |

Jumlah kredit peluncuran yang masih harus dibayar dan kredit yang diperoleh dilacak oleh metrik. CloudWatch `CPUCreditBalance` Untuk informasi lebih lanjut, lihat`CPUCreditBalance` dalam CloudWatch tabel metrik.

Contoh mode standar untuk instans yang dapat meledak

Contoh berikut menjelaskan penggunaan kredit saat instans dikonfigurasi sebagai `standard`.

Contoh

- Contoh 1: Menjelaskan penggunaan kredit dengan T3 Standa
- Contoh 2: Menjelaskan penggunaan kredit dengan T2 Standar

Contoh 1: Menjelaskan penggunaan kredit dengan T3 Standa

Dalam contoh ini, Anda melihat cara instans `t3.nano` yang diluncurkan sebagai `standard` memperoleh, mengakumulasi, dan menggunakan kredit yang diperoleh. Anda melihat cara saldo kredit mencerminkan kredit yang diperoleh yang diakumulasi.

Instans `t3.nano` yang berjalan memperoleh 144 kredit setiap 24 jam. Batas saldo kreditnya adalah 144 kredit yang diperoleh. Setelah batas tercapai, kredit baru yang diperoleh akan dibuang. Untuk informasi selengkapnya tentang jumlah kredit yang dapat diperoleh dan diakumulasi, lihat tabel kredit.

Anda dapat meluncurkan instans T3 Standar dan segera menggunakannya. Atau, Anda dapat meluncurkan instans T3 Standar dan membiarkannya idle selama beberapa hari sebelum menjalankan aplikasi di dalamnya. Digunakan atau tidaknya suatu instans akan menentukan apakah kredit akan digunakan atau diakumulasi. Jika sebuah instans tetap idle selama 24 jam sejak diluncurkan, saldo kredit mencapai batasnya, yang merupakan jumlah maksimum kredit yang diperoleh yang dapat diakumulasi.

Contoh ini menjelaskan sebuah instans yang tetap diam selama 24 jam sejak diluncurkan, dan memandu Anda melalui tujuh periode waktu selama periode 96 jam, yang menunjukkan tingkat di mana kredit diperoleh, diperoleh, digunakan, dan dibuang, serta nilai saldo kredit pada setiap akhir periode.

Alur kerja berikut mereferensikan titik-titik bernomor pada grafik:

P1 - Pada 0 jam pada grafik, instans diluncurkan sebagai `standard` dan langsung mulai mendapatkan kredit. Instans tetap diam sejak diluncurkan, yang artinya pemakaian CPU 0%, sehingga tidak ada kredit yang digunakan. Semua kredit yang tidak terpakai diakumulasi ke dalam saldo kredit. Selama 24 jam pertama, `CPUCreditUsage` berada di 0, dan nilai `CPUCreditBalance` mencapai maksimum 144.

P2 - Untuk 12 jam ke depan, pemanfaatan CPU berada pada 2,5%, yang masih di bawah acuan 5%. Instans mendapatkan lebih banyak kredit daripada yang digunakan, tetapi nilai `CPUCreditBalance` tidak dapat melebihi maksimum 144 kredit. Setiap kredit yang diperoleh yang melebihi batas akan dibuang.

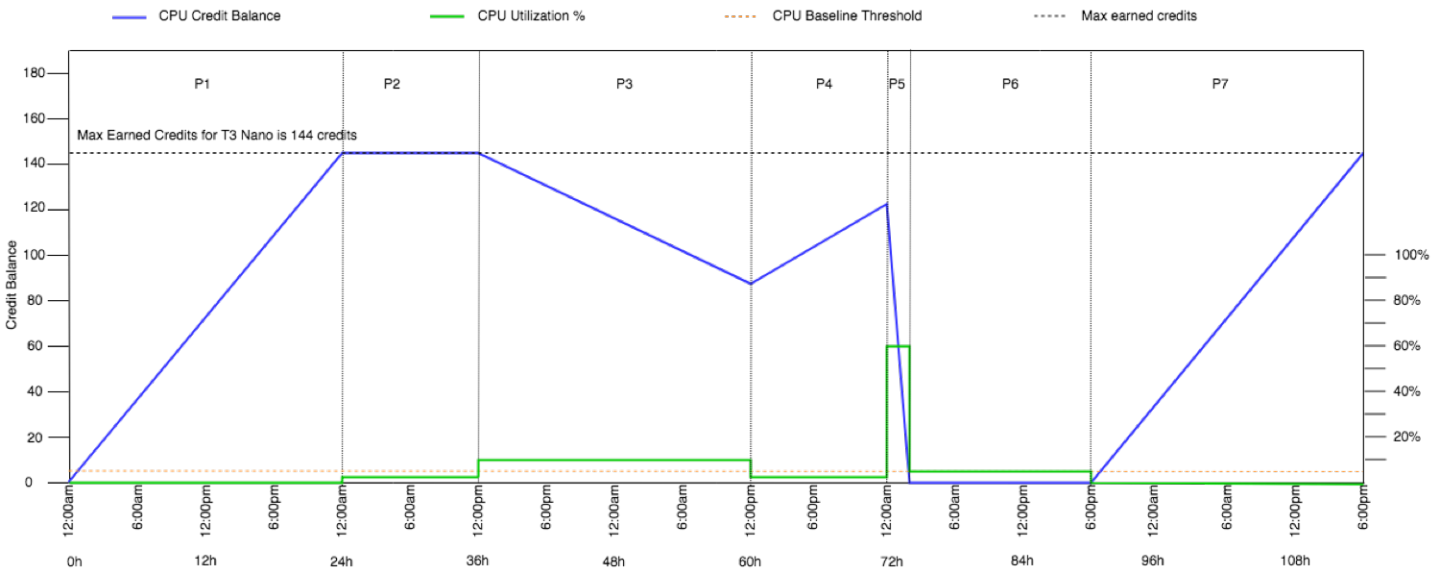
P3 - Untuk 24 jam ke depan, penggunaan CPU berada pada 7% (di atas acuan), yang membutuhkan penggunaan 57,6 kredit. Instans menggunakan lebih banyak kredit daripada yang diperolehnya, dan nilai `CPUCreditBalance` berkurang menjadi 86,4 kredit.

P4 - Selama 12 jam ke depan, pemanfaatan CPU menurun jadi 2,5% (di bawah acuan), yang membutuhkan penggunaan 36 kredit. Pada saat yang sama, instans tersebut mendapatkan 72 kredit. Instans mendapatkan lebih banyak kredit daripada yang digunakan, dan nilai `CPUCreditBalance` meningkat menjadi 122 kredit.

P5 - Selama dua jam berikutnya, instans melonjak pada pemanfaatan 60% CPU, dan menghabiskan keseluruhan nilai `CPUCreditBalance` sebesar 122 kredit. Di akhir periode ini, dengan `CPUCreditBalance` berada di nol, pemanfaatan CPU dipaksa turun ke tingkat pemanfaatan garis dasar 5%. Pada garis dasar, instans mendapatkan kredit sebanyak yang digunakan.

P6 – Untuk 14 jam ke depan, pemanfaatan CPU berada pada 5% (garis dasar). Instans ini mendapatkan kredit sebanyak yang digunakan. Nilai `CPUCreditBalance` tetap 0.

P7 - Selama 24 jam terakhir dalam contoh ini, instans tidak aktif dan pemanfaatan CPU adalah 0%. Selama waktu ini, instans mendapatkan 144 kredit, yangn diakumulasi di `CPUCreditBalance`.




Contoh 2: Menjelaskan penggunaan kredit dengan T2 Standar

Dalam contoh ini, Anda melihat cara instans `t2.nano` yang diluncurkan sebagai `standard` memperoleh, mengakumulasi, dan menggunakan kredit peluncuran dan yang diperoleh. Anda melihat cara saldo kredit mencerminkan tidak hanya kredit yang diperoleh yang diakumulasi, tetapi juga kredit peluncuran diakumulasi.

Sebuah instans `t2.nano` mendapat 30 kredit peluncuran saat diluncurkan, dan mendapatkan 72 kredit setiap 24 jam. Batas saldo kreditnya adalah 72 kredit yang diperoleh; kredit peluncuran tidak dihitung dalam batasan tersebut. Setelah batas tercapai, kredit baru yang diperoleh akan dibuang. Untuk informasi selengkapnya tentang jumlah kredit yang dapat diperoleh dan diakumulasi, lihat tabel kredit. Untuk informasi selengkapnya tentang batasn, lihat Batas kredit peluncuran.

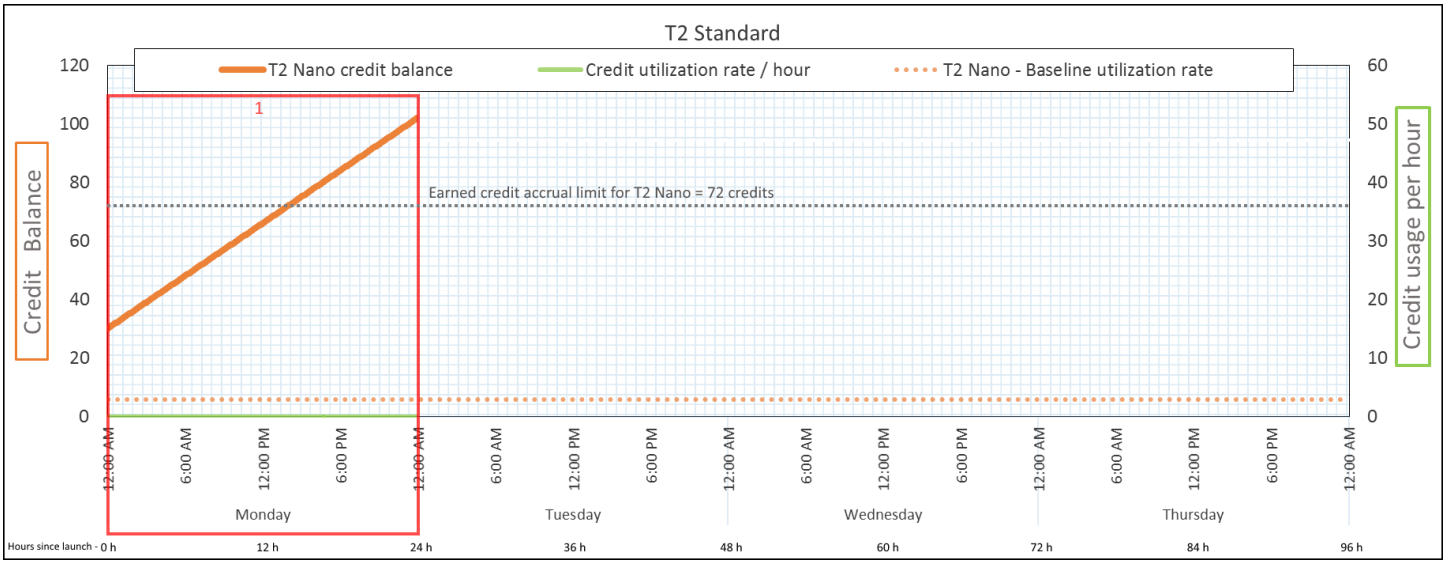
Anda dapat meluncurkan instans T2 Standar dan segera menggunakannya. Atau, Anda dapat meluncurkan instans T2 Standar dan membiarkannya idle selama beberapa hari sebelum menjalankan aplikasi di dalamnya. Digunakan atau tidaknya suatu instans akan menentukan apakah kredit akan digunakan atau diakumulasi. Jika sebuah instans tetap idle selama 24 jam sejak diluncurkan, saldo kredit tampak melebihi batasnya karena saldo tersebut mencerminkan kredit yang diperoleh diakumulasi dan kredit peluncuran yang diakumulasi. Namun, setelah CPU digunakan, kredit peluncuran digunakan terlebih dahulu. Setelah itu, batas tersebut selalu mencerminkan jumlah maksimum kredit yang diperoleh yang dapat diakumulasi.

Contoh ini menjelaskan sebuah instans yang tetap diam selama 24 jam sejak diluncurkan, dan memandu Anda melalui tujuh periode waktu selama periode 96 jam, yang menunjukkan tingkat di

mana kredit diperoleh, diperoleh, digunakan, dan dibuang, serta nilai saldo kredit pada setiap akhir periode.

Periode 1: 1 – 24 jam

Pada 0 jam pada grafik, instans T2 diluncurkan sebagai `standard` dan langsung mendapat 30 kredit peluncuran. Instans ini memperoleh kredit saat dalam kondisi berjalan. Instans tetap diam sejak diluncurkan, yang artinya pemakaian CPU 0%, sehingga tidak ada kredit yang digunakan. Semua kredit yang tidak terpakai diakumulasi ke dalam saldo kredit. Sekitar 14 jam setelah peluncuran, saldo kreditnya adalah 72 (30 kredit peluncuran + 42 kredit yang diperoleh), yang setara dengan yang dapat diperoleh instans dalam 24 jam. Pada 24 jam setelah peluncuran, saldo kredit melebihi 72 kredit karena kredit peluncuran yang tidak terpakai diakumulasikan ke saldo kredit—saldo kredit adalah 102 kredit: 30 kredit peluncuran + 72 kredit yang diperoleh.




| Metrik | Nilai |
|---|---|
| Tingkat Penggunaan Kredit | 0 kredit per 24 jam (pemanfaatan CPU 0%) |
| Tingkat Perolehan Kredit | 72 kredit per 24 jam |
| Tingkat Pembuangan Kredit | 0 kredit per 24 jam |
| Saldo Kredit | 102 kredit (30 kredit peluncuran + 72 kredit yang diperoleh) |

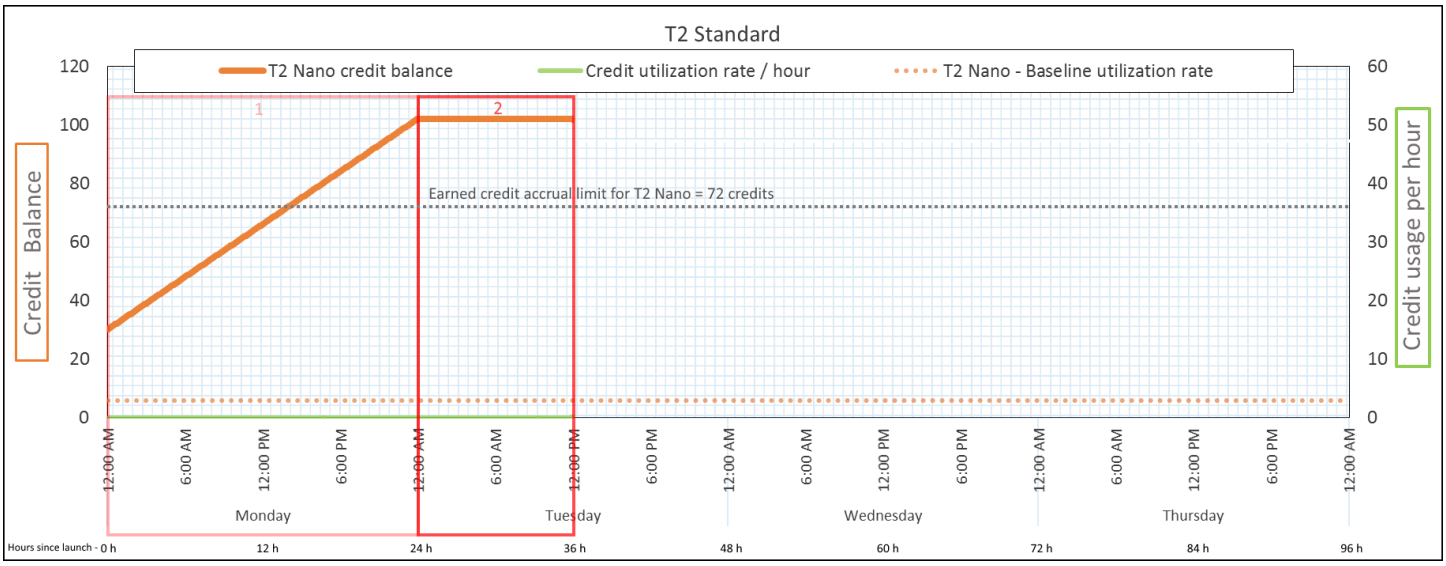
Kesimpulan

Jika tidak ada pemanfaatan CPU setelah peluncuran, instans memperoleh lebih banyak kredit daripada yang dapat diperolehnya dalam 24 jam (30 kredit peluncuran + 72 kredit yang diperoleh = 102 kredit).

Dalam skenario dunia nyata, instans EC2 menggunakan sejumlah kecil kredit saat meluncurkan dan menjalankan, yang mencegah saldo mencapai nilai teoretis maksimum dalam contoh ini.

Periode 2: 25 – 36 jam

Selama 12 jam berikutnya, instans terus idle dan memperoleh kredit, tetapi saldo kredit tidak bertambah. Saldo kredirt berhenti di 102 kredit (30 kredit peluncuran + 72 kredit yang diperoleh). Saldo kredit telah mencapai batas 72 kredit yang diperoleh yang diakumulasi, sehingga kredit yang baru diperoleh akan dibuang.



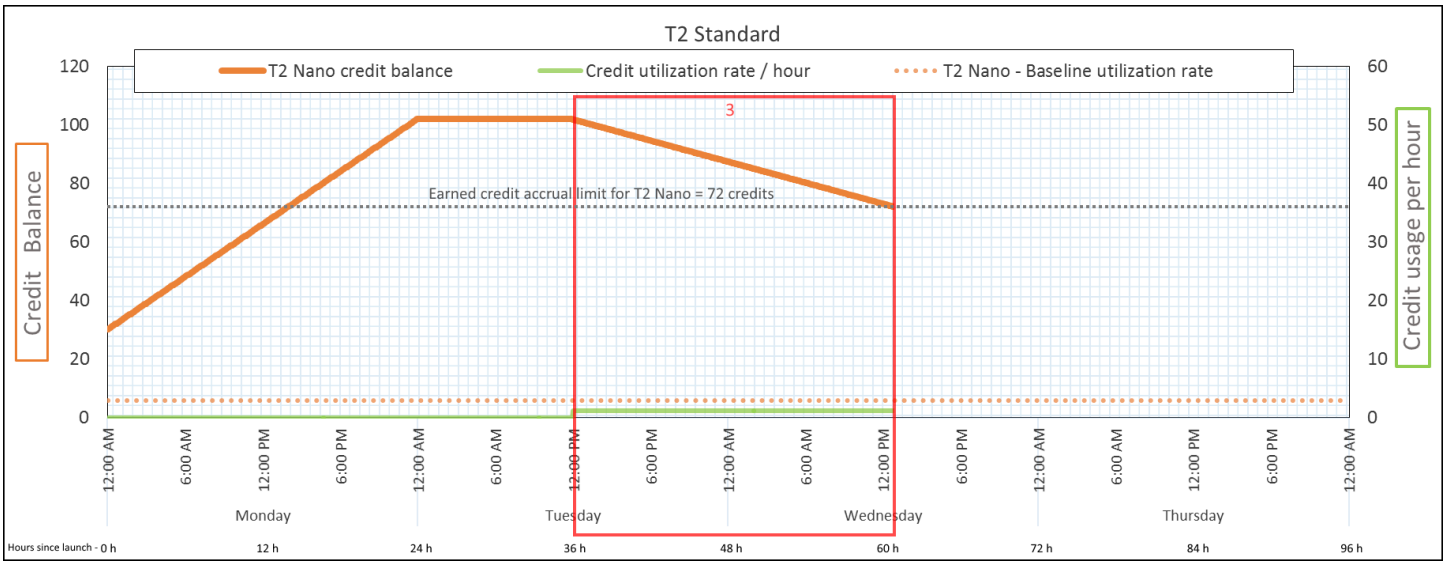

| Metrik | Nilai |
|---|---|
| Tingkat Penggunaan Kredit | 0 kredit per 24 jam (pemanfaatan CPU 0%) |
| Tingkat Perolehan Kredit | 72 kredit per 24 jam (3 kredit per jam) |
| Tingkat Pembuangan Kredit | 72 kredit per 24 jam (100% dari tingkat perolehan kredit) |
| Saldo Kredit | 102 kredit (30 kredit peluncuran + 72 kredit yang diperoleh)—saldo tidak berubah |

Kesimpulan

Sebuah instans terus-menerus memperoleh kredit, tetapi tidak dapat mengakumulasi lebih banyak kredit yang diperoleh jika saldo kredit telah mencapai batasnya. Setelah batasan tercapai, kredit yang baru diperoleh akan dibuang. Kredit peluncuran tidak termasuk dalam penghitungan batasan saldo kredit. Jika saldo termasuk kredit peluncuran yang diakumulasi, saldo tersebut tampak melebihi batas.

Periode 3: 37 – 61 jam

Selama 25 jam ke depan, instans menggunakan 2% CPU, yang membutuhkan 30 kredit. Pada periode yang sama memperoleh 75 kredit, tetapi saldo kredit menurun. Saldo menurun karena kredit peluncuran yang diakumulasi digunakan terlebih dahulu, sementara kredit yang baru diperoleh dibuang karena saldo kredit sudah mencapai batasan 72 kredit yang diperoleh.




| Metrik | Nilai |
| --- | --- |
| Tingkat Penggunaan Kredit | 28,8 kredit per 24 jam (1,2 kredit per jam, 2% pemanfaatan CPU, 40% dari tingkat perolehan kredit) —30 kredit selama 25 jam |
| Tingkat Perolehan Kredit | 72 kredit per 24 jam |
| Tingkat Pembuangan Kredit | 72 kredit per 24 jam (100% dari tingkat perolehan kredit) |

| Metrik | Nilai |
| --- | --- |
| Saldo Kredit | 72 kredit (30 kredit peluncuran digunakan; 72 kredit yang diperoleh tetap tidak digunakan) |

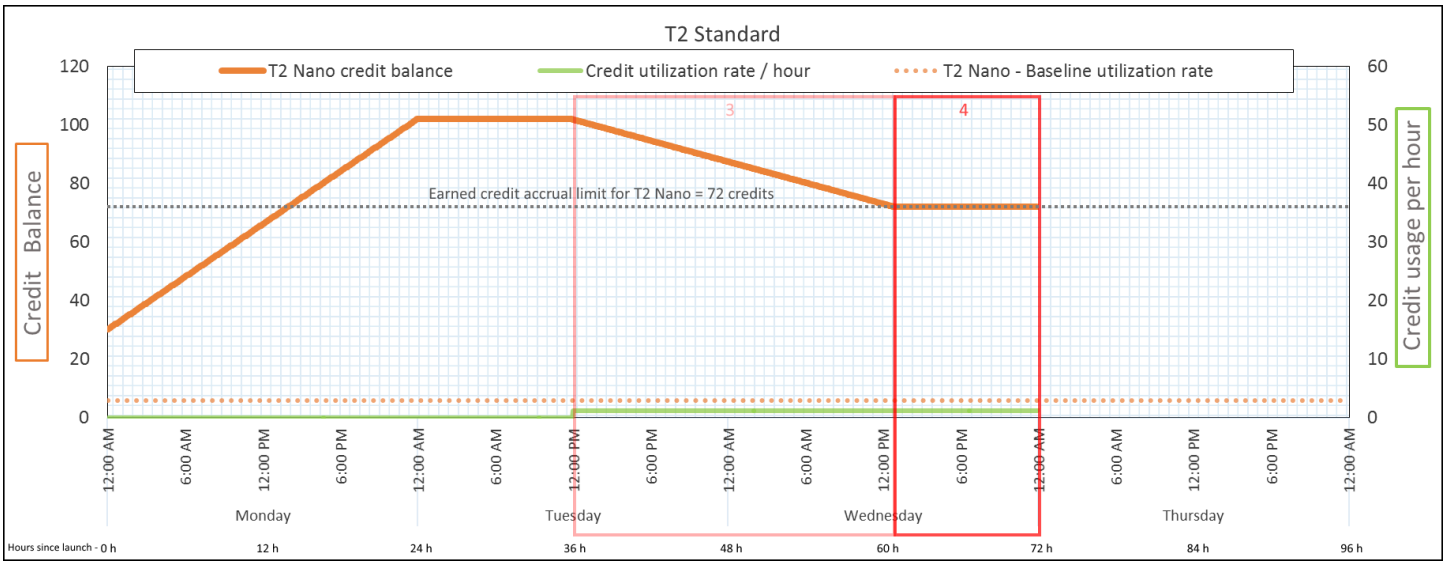
Kesimpulan

Sebuah instans menggunakan kredit peluncuran terlebih dahulu, sebelum menggunakan kredit yang diperoleh. Kredit peluncuran tidak termasuk dalam penghitungan batasan kredit. Setelah kredit peluncuran digunakan, saldonya tidak akan melebihi yang bisa diperoleh dalam 24 jam. Selain itu, saat berjalan, sebuah instans tidak dapat memperoleh lebih banyak kredit peluncuran.

Periode 4: 62 – 72 jam

Selama 11 jam ke depan, instans menggunakan 2% CPU, yang membutuhkan 13,2 kredit. Ini adalah pemanfaatan CPU yang sama seperti pada periode sebelumnya, tetapi saldo tidak berkurang. Saldo tetap berada di 72 kredit.

Saldo tidak berkurang karena tingkat pendapatan kredit lebih tinggi daripada tingkat penggunaan kredit. Saat instans menghabiskan 13,2 kredit, instans ini juga memperoleh 33 kredit. Namun, batas saldonya adalah 72 kredit, jadi setiap kredit yang diperoleh yang melebihi batas tersebut akan dibuang. Saldo mencapai titik datar di 72 kredit, yang berbeda dari puncak 102 kredit selama Periode 2, karena tidak ada kredit peluncuran yang diakumulasi.



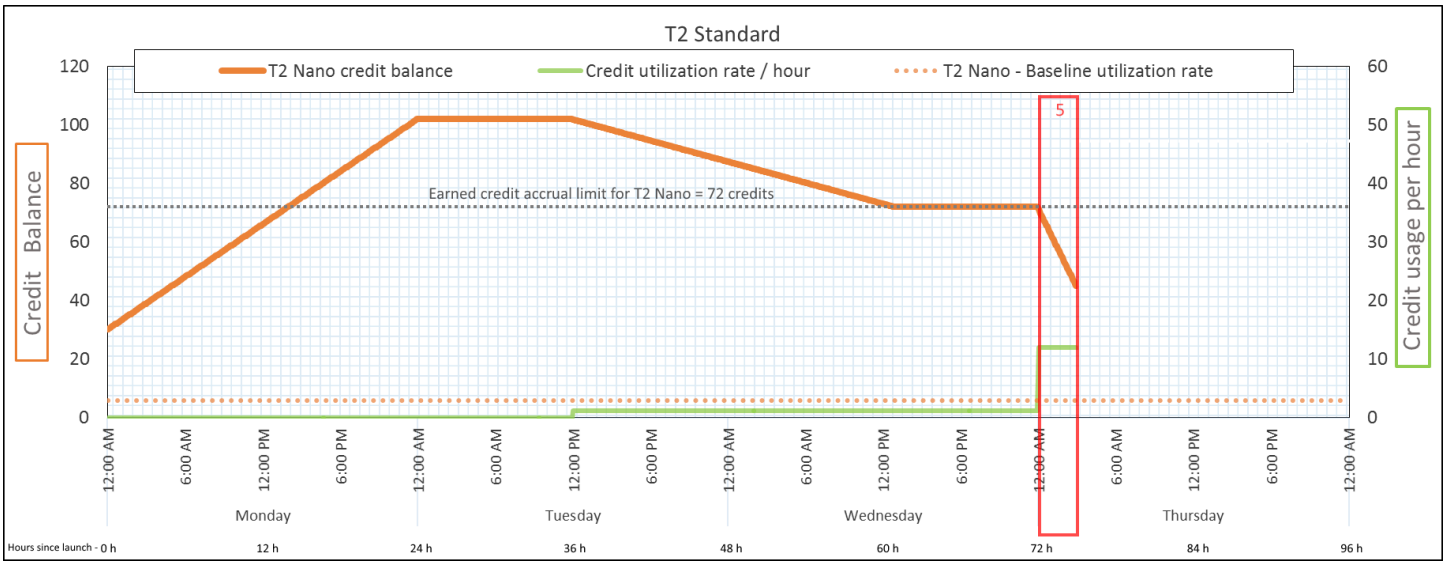
| Metrik | Nilai |
| --- | --- |
| Tingkat Penggunaan Kredit | 28,8 kredit per 24 jam (1,2 kredit per jam, 2% pemanfaatan CPU, 40% dari tingkat perolehan kredit) —13,2 kredit selama 11 jam |
| Tingkat Perolehan Kredit | 72 kredit per 24 jam |
| Tingkat Pembuangan Kredit | 43,2 kredit per 24 jam (60% dari tingkat perolehan kredit) |
| Saldo Kredit | 72 kredit (0 kredit peluncuran, 72 kredit yang diperoleh)—saldo berada pada batasnya |

Kesimpulan

Setelah kredit peluncuran digunakan, batas saldo kredit ditentukan oleh jumlah kredit yang dapat diperoleh instans dalam 24 jam. Jika instans mendapatkan lebih banyak kredit daripada yang digunakan, kredit yang baru diperoleh yang melebihi batas akan dibuang.

Periode 5: 73 – 75 jam

Selama tiga jam ke depan, instans melonjak pada 20% pemanfaatan CPU, yang membutuhkan 36 kredit. Instans ini memperoleh sembilan kredit dalam tiga jam yang sama, yang menghasilkan penurunan saldo bersih sebesar 27 kredit. Pada akhir tiga jam, saldo kredit adalah 45 kredit yang diperoleh yang diakumulasi.



| Metrik | Nilai |
| --- | --- |
| Tingkat Penggunaan Kredit | 288 kredit per 24 jam (12 kredit per jam, 20% pemanfaatan CPU, 400% dari tingkat perolehan kredit) —36 kredit selama 3 jam |
| Tingkat Perolehan Kredit | 72 kredit per 24 jam (9 kredit selama 3 jam) |
| Tingkat Pembuangan Kredit | 0 kredit per 24 jam |
| Saldo Kredit | 45 kredit (saldo sebelumnya (72) - kredit yang dihabiskan (36) + kredit yang diperoleh (9)) —saldo menurun pada tingkat 216 kredit per 24 jam (tingkat pengeluaran 288/24 + tingkat penghasilan 72/24 = tingkat 216/24 penurunan saldo) |

Kesimpulan

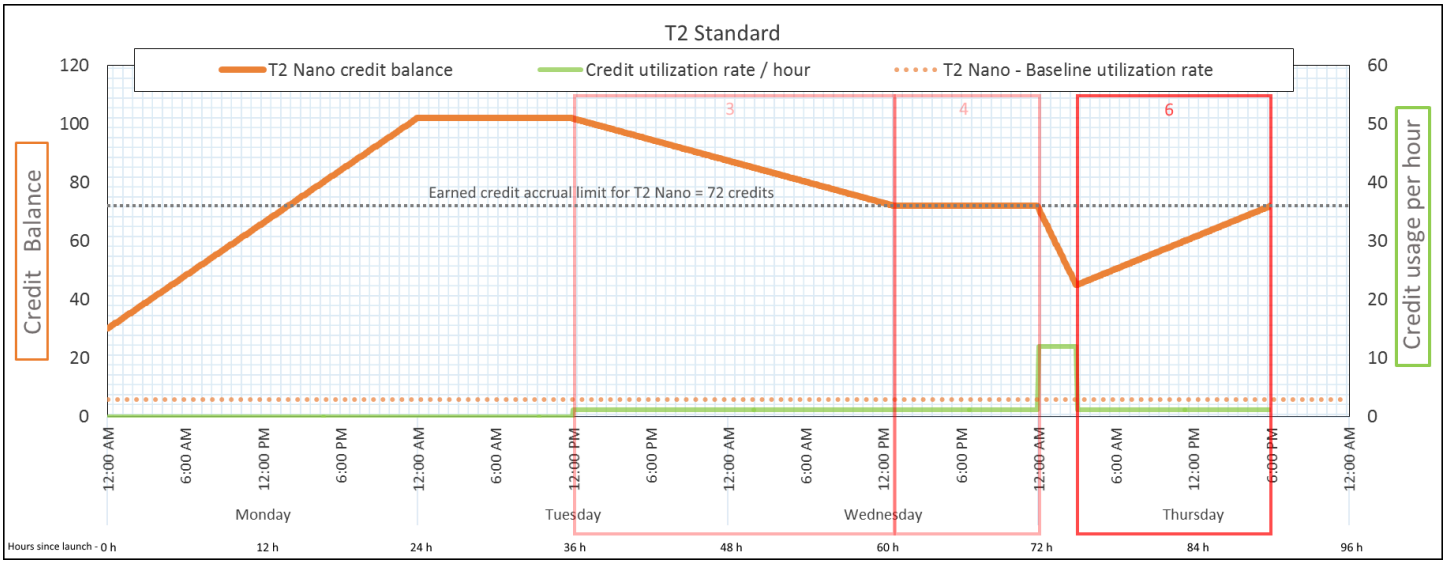
Jika sebuah instans menggunakan kredit lebih banyak daripada yang diperolehnya, saldo kreditnya menurun.

Periode 6: 76 – 90 jam

Selama 15 jam ke depan, instans menggunakan 2% CPU, yang membutuhkan 18 kredit. Ini adalah penggunaan CPU yang sama seperti pada Periode 3 dan 4. Namun, saldo meningkat pada periode ini, sedangkan pada Periode 3 menurun dan pada Periode 4 stabil.

Pada Periode 3, kredit peluncuran akumulasi digunakan, dan setiap kredit yang diperoleh yang melebihi batas kredit dibuang, yang mengakibatkan penurunan saldo kredit. Pada Periode 4, instans menggunakan lebih sedikit kredit daripada yang diperolehnya. Setiap kredit yang diperoleh yang melebihi batas dibuang, sehingga saldo menjadi stabil di maksimum 72 kredit.

Pada periode ini, tidak ada kredit peluncuran akumulasi, dan akumulasi jumlah kredit yang diperoleh dalam saldo di bawah batas. Tidak ada kredit yang diperoleh yang dibuang. Selain itu, instans tersebut mendapatkan lebih banyak kredit daripada yang digunakan, yang mengakibatkan peningkatan dalam saldo kredit.




| Metrik | Nilai |
| --- | --- |
| Tingkat Penggunaan Kredit | 28,8 kredit per 24 jam (1,2 kredit per jam, 2% pemanfaatan CPU, 40% dari tingkat perolehan kredit) —18 kredit selama 15 jam |
| Tingkat Perolehan Kredit | 72 kredit per 24 jam (45 kredit selama 15 jam) |
| Tingkat Pembuangan Kredit | 0 kredit per 24 jam |
| Saldo Kredit | 72 kredit (saldo meningkat pada tingkat 43,2 kredit per 24 jam—tingkat perubahan = tingkat pengeluaran 28. 8/24 + tingkat penghasilan 72/24) |

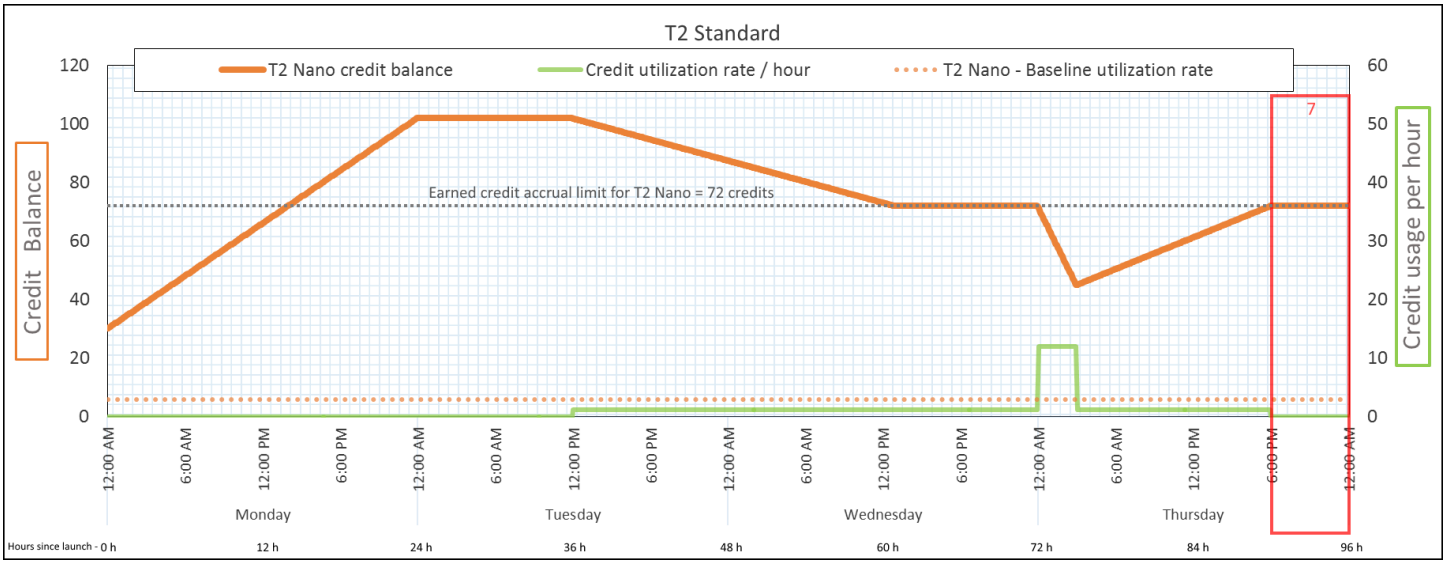
Kesimpulan

Jika sebuah instans menggunakan kredit lebih sedikit daripada yang diperolehnya, saldo kreditnya meningkat.

Periode 7: 91 – 96 jam

Selama enam jam berikutnya, instans tetap idle— pemakaian CPU 0%—dan tidak ada kredit yang digunakan. Ini adalah penggunaan CPU yang sama seperti di Periode 2, tetapi saldonya tidak berhenti di 102 kredit—saldo berhenti di 72 kredit, yang merupakan batas saldo kredit untuk instans.

Pada Periode 2, saldo kredit termasuk akumulasi 30 kredit peluncuran. Kredit peluncuran digunakan di Periode 3. Instans yang berjalan tidak bisa mendapatkan lebih banyak kredit peluncuran. Setelah batas saldo kredit CPU tercapai, kredit apa pun yang diperoleh setelah batas akan dibuang.




| Metrik | Nilai |
| --- | --- |
| Tingkat Penggunaan Kredit | 0 kredit per 24 jam (pemanfaatan CPU 0%) |
| Tingkat Perolehan Kredit | 72 kredit per 24 jam |
| Tingkat Pembuangan Kredit | 72 kredit per 24 jam (100% dari tingkat perolehan kredit) |
| Saldo Kredit | 72 kredit (0 kredit peluncuran + 72 kredit perolehan) |

## Kesimpulan

Sebuah instans terus-menerus memperoleh kredit, tetapi tidak dapat mengakumulasi lebih banyak kredit yang diperoleh jika saldo kredit telah tercapai. Setelah batasan tercapai, kredit yang baru diperoleh akan dibuang. Batas saldo kredit ditentukan oleh jumlah kredit yang dapat diperoleh instans dalam 24 jam. Untuk informasi selengkapnya tentang batas saldo kredit, lihat tabel kredit.

## Konfigurasikan instans kinerja yang dapat meledak

Langkah-langkah untuk meluncurkan, memantau, dan memodifikasi instans kinerja yang meledak (instans T) serupa. Perbedaan utamanya adalah spesifikasi kredit default saat diluncurkan.

Setiap keluarga instance T dilengkapi dengan spesifikasi kredit default berikut:

- Instans T4g, T3a, dan T3 diluncurkan sebagai `unlimited`
- Instans T3 pada Host Khusus hanya dapat diluncurkan sebagai `standard`
- Instans T2 diluncurkan sebagai `standard`

Anda dapat mengubah spesifikasi kredit default untuk akun tersebut.

Tugas

- Konfigurasikan spesifikasi kredit saat peluncuran
- Mengkonfigurasi grup Penskalaan Otomatis untuk menetapkan spesifikasi kredit sebagai tak terbatas
- Mengelola spesifikasi kredit instans kinerja yang meledak
- Mengelola spesifikasi kredit default untuk akun

### Konfigurasikan spesifikasi kredit saat peluncuran

Anda dapat meluncurkan instans T dengan spesifikasi kredit `unlimited` atau`standard`.

Prosedur berikut menjelaskan cara menggunakan konsol EC2 atau. AWS CLI Untuk informasi tentang menggunakan grup Penskalaan Otomatis, lihatMengkonfigurasi grup Penskalaan Otomatis untuk menetapkan spesifikasi kredit sebagai tak terbatas.

Console

Untuk mengkonfigurasi spesifikasi kredit instans saat peluncuran

1. Ikuti prosedur untuk meluncurkan instans.
2. Pada Tipe instans, pilih tipe instans T.
3. Perluas detail lanjutan. Untuk Spesifikasi kredit, pilih spesifikasi kredit.
4. Di panel Ringkasan, tinjau konfigurasi instans Anda, lalu pilih Luncurkan instans.

AWS CLI

Untuk mengatur spesifikasi kredit instans saat peluncuran

Gunakan perintah run-instance dengan opsi. `--credit-specification`

```
--credit-specification CpuCredits=unlimited
```

PowerShell

Untuk mengatur spesifikasi kredit instans saat peluncuran

Gunakan New-EC2Instance cmdlet dengan parameter. `-CreditSpecification_CpuCredit`

```
-CreditSpecification_CpuCredit unlimited
```

Mengkonfigurasi grup Penskalaan Otomatis untuk menetapkan spesifikasi kredit sebagai tak terbatas

Ketika instance T diluncurkan atau dimulai, mereka memerlukan kredit CPU untuk pengalaman bootstrap yang baik. Jika Anda menggunakan grup Auto Scaling untuk meluncurkan instans, sebaiknya konfigurasikan instans Anda sebagai `unlimited`. Jika Anda melakukannya, instans menggunakan surplus kredit saat diluncurkan atau dimulai ulang secara otomatis oleh grup Auto Scaling. Menggunakan kredit surplus mencegah pembatasan performa.

Membuat templat peluncuran

Anda harus menggunakan templat peluncuran untuk meluncurkan instans sebagai `unlimited` dalam grup Auto Scaling. Konfigurasi peluncuran tidak mendukung peluncuran instans sebagai `unlimited`.

Console

Untuk membuat template peluncuran yang menetapkan spesifikasi kredit

1. Ikuti prosedur Buat templat peluncuran menggunakan pengaturan lanjutan di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
2. Dalam Konten templat peluncuran, untuk Tipe instans, pilih ukuran instans.
3. Untuk meluncurkan instans sebagai `unlimited` dalam grup Auto Scaling, pada Detail lanjutan, untuk Spesifikasi kredit, pilih Tak Terbatas.

4. Setelah Anda selesai menentukan parameter templat peluncuran, pilih Buat templat peluncuran.

AWS CLI

Untuk membuat template peluncuran yang menetapkan spesifikasi kredit

Gunakan perintah create-launch-template.

```
aws ec2 create-launch-template \
    --launch-template-name my-launch-template \
    --version-description FirstVersion \
    --launch-template-data CreditSpecification={CpuCredits=unlimited}
```

PowerShell

Untuk membuat template peluncuran yang menetapkan spesifikasi kredit

Gunakan New-EC2LaunchTemplate cmdlet. Tentukan spesifikasi kredit untuk data template peluncuran sebagai berikut.

```
$creditSpec = New-Object Amazon.EC2.Model.CreditSpecificationRequest
$creditSpec.CpuCredits = "unlimited"
$launchTemplateData = New-Object Amazon.EC2.Model.RequestLaunchTemplateData
$launchTemplateData.CreditSpecification = $creditSpec
```

Kaitkan grup Auto Scaling dengan templat peluncuran

Untuk mengaitkan templat peluncuran dengan grup Auto Scaling, buat grup Auto Scaling menggunakan templat peluncuran, atau tambahkan templat peluncuran ke grup Auto Scaling yang sudah ada.

Console

Untuk membuat grup Auto Scaling menggunakan templat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi di bagian atas layar, pilih kawasan yang sama yang Anda gunakan saat Anda membuat templat peluncuran.

3. Di panel navigasi, pilih Grup Auto Scaling, pilih Buat grup Auto Scaling.
4. Pilih Templat Peluncuran, pilih templat peluncuran Anda, lalu pilih Langkah Berikutnya.
5. Lengkapi bidang grup Auto Scaling. Setelah Anda selesai meninjau pengaturan konfigurasi di halaman Pratinjau, pilih Buat grup Auto Scaling. Untuk informasi selengkapnya, lihat Membuat Group Auto Scaling Menggunakan Templat Peluncuran di Panduan Pengguna Amazon EC2 Auto Scaling.

Untuk menambahkan templat peluncuran ke grup Auto Scaling yang ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi di bagian atas layar, pilih Wilayah yang sama dengan yang Anda gunakan saat Anda membuat templat peluncuran.
3. Di panel navigasi, pilih Grup Auto Scaling.
4. Dari daftar grup Auto Scaling, pilih grup Auto Scaling, dan pilih Tindakan, Edit.
5. Pada tab Detail, untuk Templat Peluncuran, pilih templat peluncuran, lalu pilih Simpan.

AWS CLI

Untuk membuat grup Auto Scaling menggunakan templat peluncuran

Gunakan perintah create-auto-scaling-group dan tentukan parameternya. `--launch-template`

Untuk menambahkan templat peluncuran ke grup Auto Scaling yang ada

Gunakan perintah update-auto-scaling-group dan tentukan parameternya. `--launch-template`

PowerShell

Untuk membuat grup Auto Scaling menggunakan templat peluncuran

Gunakan New-ASAutoScalingGroup cmdlet dan tentukan parameter `-LaunchTemplate_LaunchTemplateId` atau `-LaunchTemplate_LaunchTemplateName`.

Untuk menambahkan templat peluncuran ke grup Auto Scaling yang ada

Gunakan Update-ASAutoScalingGroup cmdlet dan tentukan parameter `-LaunchTemplate_LaunchTemplateId` atau `-LaunchTemplate_LaunchTemplateName`.

### Mengelola spesifikasi kredit instans kinerja yang meledak

Anda dapat mengganti spesifikasi kredit dari instance T yang sedang berjalan atau dihentikan kapan saja antara `unlimited` dan`standard`.

Perhatikan bahwa dalam mode `unlimited`, sebuah instans dapat menghabiskan kredit surplus, yang mungkin menimbulkan biaya tambahan. Untuk informasi selengkapnya, lihat Kredit surplus dapat dikenakan biaya.

#### Console

Untuk mengelola spesifikasi kredit suatu instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. (Opsional) Pilih instance. Pada tab Detailnya, temukan Spesifikasi Kredit. Nilainya adalah `unlimited` atau `standard`.
4. (Opsional) Untuk mengubah spesifikasi kredit untuk beberapa instans secara bersamaan, pilih semuanya.
5. Pilih Tindakan, Pengaturan instans, Ubah spesifikasi kredit. Opsi ini diaktifkan hanya jika Anda memilih instance T.
6. Untuk mode Unlimited, pilih atau kosongkan kotak centang di samping setiap ID instance.

#### AWS CLI

Untuk mendapatkan spesifikasi kredit suatu instans

Gunakan perintah describe-instance-credit-specifications. Jika Anda tidak menentukan ID instance, semua instance dengan spesifikasi kredit `unlimited` dikembalikan. Output juga akan mencakup contoh yang sebelumnya dikonfigurasi dengan spesifikasi `unlimited` kredit. Misalnya, jika Anda mengubah ukuran instans T3 menjadi instans M4, saat dikonfigurasi sebagai, `unlimited`Amazon EC2 mengembalikan instans M4.

```
aws ec2 describe-instance-credit-specifications \
    --instance-id i-1234567890abcdef0 \
    --query InstanceCreditSpecifications[].CpuCredits \
    --output text
```

Berikut ini adalah output contoh.

```
unlimited
```

Untuk mengatur spesifikasi kredit suatu instans

Gunakan perintah modify-instance-credit-specification.

```
aws ec2 modify-instance-credit-specification \
    --region us-east-1 \
    --instance-credit-specification
 "InstanceId=i-1234567890abcdef0,CpuCredits=unlimited"
```

PowerShell

Untuk mendapatkan spesifikasi kredit suatu instans

Gunakan Get-EC2CreditSpecification cmdlet.

```
(Get-EC2CreditSpecification `
    -InstanceId i-1234567890abcdef0).CpuCredits
```

Berikut ini adalah output contoh.

```
unlimited
```

Untuk mengatur spesifikasi kredit suatu instans

Gunakan Edit-EC2InstanceCreditSpecification cmdlet.

```
Edit-EC2InstanceCreditSpecification `
    -Region us-east-1 `
    -InstanceCreditSpecification @({InstanceId="i-1234567890abcdef0"
 CpuCredits="unlimited"})
```

### Mengelola spesifikasi kredit default untuk akun

Setiap keluarga instance T dilengkapi dengan spesifikasi kredit default. Anda dapat mengubah spesifikasi kredit default untuk setiap keluarga instans T di tingkat akun per Wil AWS ayah. Nilai yang valid untuk spesifikasi kredit default adalah `unlimited` dan`standard`.

Jika Anda menggunakan wizard peluncuran instans di konsol EC2 untuk meluncurkan instans, nilai yang Anda pilih untuk spesifikasi kredit akan menggantikan spesifikasi kredit default tingkat akun.

Jika Anda menggunakan AWS CLI untuk meluncurkan instans, semua instans T baru di akun akan diluncurkan menggunakan spesifikasi kredit default. Spesifikasi kredit untuk instans yang sedang berjalan atau dihentikan tidak terpengaruh.

Pertimbangan

Spesifikasi kredit default untuk keluarga instans hanya dapat dimodifikasi sekali dalam periode 5 menit bergulir, dan hingga empat kali dalam periode 24 jam bergulir.

Console

Untuk mengelola spesifikasi kredit default

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Untuk mengubah Wilayah AWS, gunakan pemilih Wilayah di sudut kanan atas halaman.
3. Di panel navigasi, pilih Dasbor.
4. Pada kartu atribut akun, di bawah Peng aturan, pilih Spesifikasi kredit default.
5. Pilih Kelola.
6. Untuk setiap keluarga instans, pilih Tak Terbatas atau Standard, lalu pilih Perbarui.

AWS CLI

Untuk mendapatkan spesifikasi kredit default

Gunakan perintah get-default-credit-specification.

```
aws ec2 get-default-credit-specification \
    --region us-east-1 \
    --instance-family t2 \
    --query InstanceFamilyCreditSpecifications[].CpuCredits \
    --output text
```

Berikut ini adalah output contoh.

```
standard
```

Untuk mengatur spesifikasi kredit default

Gunakan perintah modify-default-credit-specification. Contoh berikut menetapkan nilai keunlimited.

```
aws ec2 modify-default-credit-specification \
    --region us-east-1 \
    --instance-family t2 \
    --cpu-credits unlimited
```

PowerShell

Untuk mendapatkan spesifikasi kredit default

Gunakan Get-EC2DefaultCreditSpecification cmdlet.

```
(Get-EC2DefaultCreditSpecification `
    -Region us-east-1 `
    -InstanceFamily t2).CpuCredits
```

Berikut ini adalah output contoh.

```
standard
```

Untuk mengatur spesifikasi kredit default

Gunakan Edit-EC2DefaultCreditSpecification cmdlet. Contoh berikut menetapkan nilai keunlimited.

```
Edit-EC2DefaultCreditSpecification `
    -Region us-east-1 `
    -InstanceFamily t2 `
    -CpuCredit unlimited
```

## Memantau kredit CPU untuk instans yang dapat meledak

EC2 mengirimkan metrik ke Amazon CloudWatch. Anda dapat melihat metrik kredit CPU di metrik per instans Amazon EC2 di CloudWatch konsol atau dengan menggunakan AWS CLI untuk membuat daftar metrik untuk setiap instans. Untuk informasi selengkapnya, lihat CloudWatch metrik yang tersedia untuk instans Anda.

Daftar Isi


- CloudWatch Metrik tambahan untuk instans kinerja yang dapat meledak
- Menghitung penggunaan kredit CPU

CloudWatch Metrik tambahan untuk instans kinerja yang dapat meledak

Instans kinerja burstable memiliki CloudWatch metrik tambahan ini, yang diperbarui setiap lima menit:

- `CPUCreditUsage` – Jumlah kredit CPU yang digunakan selama periode pengukuran.
- `CPUCreditBalance` – Jumlah kredit CPU yang diakumulasi oleh instans. Saldo ini habis saat CPU melonjak dan kredit CPU digunakan lebih cepat daripada yang diperoleh.
- `CPUSurplusCreditBalance` – Jumlah kredit CPU surplus yang digunakan untuk mempertahankan pemanfaatan CPU saat nilai `CPUCreditBalance` adalah nol.
- `CPUSurplusCreditsCharged` – Jumlah kredit CPU surplus yang melebihi jumlah kredit CPU maksimum yang dapat diperoleh dalam periode 24 jam, dan dengan demikian menarik biaya tambahan.

Dua metrik terakhir hanya berlaku untuk instans yang dikonfigurasi sebagai `unlimited`.

Tabel berikut menjelaskan CloudWatch metrik untuk instans kinerja yang dapat meledak. Untuk informasi selengkapnya, lihat CloudWatch metrik yang tersedia untuk instans Anda.

| Metrik | Deskripsi |
| --- | --- |
| `CPUCreditUsage` | Jumlah kredit CPU yang digunakan oleh instans untuk pemanfaatan CPU. Satu kredit CPU sama dengan satu vCPU yang berjalan dengan pemanfaatan 100% selama satu menit atau kombinasi yang setara dari vCPU, pemanfaatan, dan waktu (misalnya, satu vCPU yang berjalan dengan pemanfaat an 50% selama dua menit atau dua vCPU yang berjalan dengan pemanfaatan 25% selama dua menit).<br><br>Metrik kredit CPU tersedia pada frekuensi lima menit saja. Jika Anda menentukan periode lebih dari lima menit, gunakan statistik `Sum`, bukan statistik `Average`.<br><br>Unit: Kredit (vCPU-minutes) |
| `CPUCreditBalance` | Jumlah kredit CPU yang diperoleh yang diakumulasi oleh instans sejak diluncurkan atau dimulai. Untuk T2 Standar, `CPUCreditBalance` juga mencakup jumlah kredit peluncuran yang telah diakumulasi. |

| Metrik | Deskripsi |
| --- | --- |
| | Kredit diakumulasi ke saldo kredit setelah diperoleh, dan dihapus dari saldo kredit saat digunakan. Saldo kredit memiliki batas maksimum, yang ditentukan oleh ukuran instans. Setelah batas tercapai, setiap kredit yang baru diperoleh akan dibuang. Untuk T2 Standar, kredit peluncuran tidak termasuk dalam penghitungan batas.<br><br>Kredit dalam `CPUCreditBalance` tersedia untuk instans untuk digunakan hingga melonjak melebihi pemanfaatan CPU acuan.<br><br>Saat sebuah instans berjalan, kredit di `CPUCreditBalance` tidak kedaluarsa. Ketika instance T4g, T3a atau T3 berhenti, nil `CPUCreditBalance` ainya bertahan selama tujuh hari. Setelah itu, semua kredit akumulasi akan hilang. Saat instans T2 berhenti, nilai `CPUCreditBalance` tidak bertahan, dan semua kredit akumulasi akan hilang.<br><br>Metrik kredit CPU tersedia pada frekuensi lima menit saja.<br><br>Unit: Kredit (vCPU-minutes) |
| `CPUSurplusCreditBalance` | Jumlah kredit surplus yang telah digunakan oleh instans `unlimited` saat nilai `CPUCreditBalance` miliknya adalah nol.<br><br>Nilai `CPUSurplusCreditBalance` dibayarkan oleh dengan kredit CPU yang diperoleh. Jika jumlah kredit surplus melebihi jumlah kredit maksimum yang dapat diperoleh instans dalam jangka waktu 24 jam, kredit surplus yang digunakan di atas jumlah maksimum akan dikenai biaya tambahan.<br><br>Unit: Kredit (vCPU-minutes) |

| Metrik | Deskripsi |
| --- | --- |
| `CPUSurplusCreditsCharged` | Jumlah kredit surplus yang digunakan yang tidak ditutupi oleh kredit CPU yang diperoleh, sehingga menimbulkan biaya tambahan.<br><br>Kredit surplus yang digunakan akan dikenai biaya jika salah satu dari hal berikut terjadi:<br><br>• Kredit surplus yang digunakan melampaui jumlah kredit maksimum yang bisa didapatkan oleh instans dalam periode 24 jam. Kredit surplus yang digunakan di atas jumlah maksimum akan dikenai biaya pada akhir jam.<br>• Instans dihentikan atau diakhiri.<br>• instans dialihkan dari `unlimited` ke `standard`.<br><br>Unit: Kredit (vCPU-minutes) |

Menghitung penggunaan kredit CPU

Penggunaan kredit CPU instans dihitung menggunakan CloudWatch metrik instans yang dijelaskan dalam tabel sebelumnya.

Amazon EC2 mengirimkan metrik ke CloudWatch setiap lima menit. Referensi ke nilai sebelumnya dari metrik pada titik waktu mana pun menyiratkan nilai sebelumnya dari metrik, yang dikirimkan lima menit yang lalu.

Menghitung penggunaan kredit CPU untuk instans Standar

- Saldo kredit CPU meningkat jika pemanfaatan CPU di bawah garis dasar, ketika kredit yang digunakan kurang dari kredit yang diakumulasi dalam interval lima menit sebelumnya.
- Saldo kredit CPU berkurang jika pemakaian CPU di atas garis dasar, ketika kredit yang digunakan lebih dari kredit yang diperoleh dalam interval lima menit sebelumnya.

Secara matematis, hal tersebut ditangkap oleh persamaan berikut:

Example

```
CPUCreditBalance = prior CPUCreditBalance + [Credits earned per hour * (5/60) -
 CPUCreditUsage]
```

Ukuran instans menentukan jumlah kredit yang dapat diperoleh instans per jam dan jumlah kredit yang diperoleh yang dapat diakumulasi dalam saldo kredit. Untuk informasi tentang jumlah kredit yang diperoleh per jam, dan batas saldo kredit untuk setiap ukuran instans, lihat tabel kredit.

Contoh

Contoh ini menggunakan instans `t3.nano`. Untuk menghitung nilai `CPUCreditBalance` instans, gunakan persamaan sebelumnya sebagai berikut:

- `CPUCreditBalance` – Saldo kredit saat ini yang akan dihitung.
- `prior CPUCreditBalance` – Saldo kredit lima menit lalu. Dalam contoh ini, instans telah mengakumulasi dua kredit.
- `Credits earned per hour` – Sebuah instans `t3.nano` memperoleh enam kredit per jam.
- `5/60`- Mewakili interval lima menit antara CloudWatch publikasi metrik. Kalikan kredit yang diperoleh per jam dengan 5/60 (lima menit) untuk mendapatkan jumlah kredit yang diperoleh instans dalam lima menit terakhir. Instans `t3.nano` memperoleh 0,5 kredit setiap lima menit.
- `CPUCreditUsage` – Banyaknya kredit yang digunakan instans dalam lima menit terakhir. Dalam contoh ini, instans menggunakan satu kredit dalam lima menit terakhir.

Dengan menggunakan nilai-nilai ini, Anda dapat menghitung nilai `CPUCreditBalance`:

Example

```
CPUCreditBalance = 2 + [0.5 - 1] = 1.5
```

Menghitung penggunaan kredit CPU untuk instans Tidak Terbatas

Ketika instans performa yang dapat melonjak perlu melonjak di atas garis dasar, instans akan menggunakan kredit yang diakumulasi sebelum mengunakan kredit surplus. Saat menggunakan saldo kredit CPU yang diakumulasi, instans dapat menggunakan kredit surplus untuk melonjakkan CPU selama yang dibutuhkannya. Saat pemanfaatan CPU turun di bawah garis dasar, kredit surplus akan dibayarkan sebelum instans mengakumulasi kredit yang diperoleh.

Kami menggunakan istilah `Adjusted balance` dalam persamaan berikut untuk mencerminkan aktivitas yang terjadi dalam interval lima menit ini. Kami menggunakan nilai ini untuk sampai pada nilai untuk `CPUSurplusCreditBalance` CloudWatch metrik `CPUCreditBalance` dan.

Example

```
Adjusted balance = [prior CPUCreditBalance - prior CPUSurplusCreditBalance] + [Credits
 earned per hour * (5/60) - CPUCreditUsage]
```

Nilai `0` untuk `Adjusted balance` menunjukkan bahwa instans menggunakan semua kredit yang diperoleh untuk melonjak, dan tidak ada kredit surplus yang digunakan. Hasilnya, baik `CPUCreditBalance` dan `CPUSurplusCreditBalance` diatur ke `0`.

Nilai `Adjusted balance` positif menunjukkan bahwa kredit yang diperoleh yang diakumulasi oleh instans, dan kredit surplus sebelumnya, jika ada, telah dibayarkan. Oleh karena itu, nilai `Adjusted balance` ditetapkan ke `CPUCreditBalance` dan `CPUSurplusCreditBalance` diatur ke `0`. Ukuran instans menentukan jumlah kredit maksimum yang dapat diperoleh.

Example

```
CPUCreditBalance = min [max earned credit balance, Adjusted balance]
CPUSurplusCreditBalance = 0
```

Nilai `Adjusted balance` negatif menunjukkan bahwa instans menggunakan semua kredit yang diperoleh yang diakumulasi dan, selain itu, juga menggunakan kredit surplus untuk melonjak. Oleh karena itu, nilai `Adjusted balance` ditetapkan ke `CPUSurplusCreditBalance` dan `CPUCreditBalance` diatur ke `0`. Sekali lagi, ukuran instans menentukan jumlah kredit maksimum yang dapat diakumulasikan.

Example

```
CPUSurplusCreditBalance = min [max earned credit balance, -Adjusted balance]
CPUCreditBalance = 0
```

Jika kredit surplus yang digunakan melebihi kredit maksimum yang dapat diakumulasi oleh instans, saldo kredit surplus diatur ke maksimum, seperti yang ditunjukkan dalam persamaan sebelumnya. Kredit surplus yang tersisa dikenakan tagihan sebagaimana direpresentasikan oleh metrik `CPUSurplusCreditsCharged`.

Example

```
CPUSurplusCreditsCharged = max [-Adjusted balance - max earned credit balance, 0]
```

Akhirnya, saat instans berakhir, semua kredit surplus yang dilacak oleh `CPUSurplusCreditBalance` dikenakan tagihan. Jika instans dialihkan dari `unlimited` ke `standard`, setiap `CPUSurplusCreditBalance` yang tersisa juga dikenakan biaya.

## Akselerasi kinerja dengan instans GPU

GPU-based instance menyediakan akses ke GPU NVIDIA dengan ribuan inti komputasi. Anda dapat menggunakan instans ini untuk mengakselerasi aplikasi ilmiah, rekayasa, dan rendering dengan memanfaatkan kerangka kerja komputasi paralel CUDA atau Open Computing Language (OpenCL). Anda juga dapat menggunakannya untuk aplikasi grafik, termasuk streaming game, streaming aplikasi 3-D, dan beban kerja grafis lainnya.

Sebelum Anda dapat mengaktifkan atau mengoptimalkan GPU-based instance, Anda harus menginstal driver yang sesuai, sebagai berikut:

- Untuk menginstal driver NVIDIA pada instance dengan GPU NVIDIA terlampir, seperti instans P3 atau G4dn, lihat. Driver NVIDIA
- Untuk menginstal driver AMD pada instance dengan GPU AMD terlampir, seperti instance G4ad, lihat. Driver AMD

Daftar Isi


- Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda
- Optimalkan pengaturan GPU pada instans Amazon EC2
- Mengatur tampilan 4K ganda pada instans G4ad Linux
- Memulai dengan instans yang dipercepat GPU


### Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda

Untuk mengaktifkan Aplikasi Virtual GRID pada GPU-based instans yang memiliki GPU NVIDIA (NVIDIA GRID Virtual Workstation diaktifkan secara default), Anda harus menentukan jenis produk untuk driver. Proses yang Anda gunakan tergantung pada sistem operasi instans Anda.

### Instans Linux

Untuk mengaktifkan Aplikasi Virtual GRID pada instans Linux Anda

1. Buat file `/etc/nvidia/gridd.conf` dari file templat yang disediakan.

   ```
   [ec2-user ~]$ sudo cp /etc/nvidia/gridd.conf.template /etc/nvidia/gridd.conf
   ```

2. Buka file `/etc/nvidia/gridd.conf` dalam editor teks favorit Anda.
3. Temukan baris `FeatureType`, dan atur ke `0`. Kemudian, tambahkan baris dengan `IgnoreSP=TRUE`.

   ```
   FeatureType=0 IgnoreSP=TRUE
   ```

4. Simpan file dan keluar.
5. Mulai ulang instans untuk mengambil konfigurasi baru.

   ```
   [ec2-user ~]$ sudo reboot
   ```

### Instans Windows

Untuk mengaktifkan Aplikasi Virtual GRID pada instans Windows Anda

1. Jalankan regedit.exe untuk membuka editor registri.
2. Navigasi ke `HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global \GridLicensing`.
3. Buka menu konteks (klik kanan) di panel kanan dan pilih Baru, WORD.
4. Untuk Nama , masukkanFeatureType dan ketik`Enter` .
5. Buka menu konteks (klik kanan) diFeatureType dan pilih Modifikasi .
6. Pada Data nilai, masukkan `0` untuk Aplikasi Virtual NVIDIA GRID dan pilih OK.
7. Buka menu konteks (klik kanan) di panel kanan dan pilih Baru, WORD.
8. Untuk Nama, masukkan IgnoreSP dan ketik `Enter`.
9. Buka menu konteks (klik kanan) pada IgnoreSP dan pilih Modifikasi.
10. Untuk Nilai data, ketik 1 dan pilih OK.
11. Tutup editor registri.

# Optimalkan pengaturan GPU pada instans Amazon EC2

Ada beberapa pengoptimalan pengaturan GPU yang dapat Anda lakukan untuk mencapai performa terbaik pada instans GPU NVIDIA. Dengan beberapa tipe instans ini, driver NVIDIA menggunakan fitur lonjak otomatis, yang memvariasikan kecepatan clock GPU. Dengan menonaktifkan autoboost dan menyetel kecepatan clock GPU ke frekuensi maksimumnya, Anda dapat secara konsisten mencapai kinerja maksimum dengan instans GPU Anda.

Optimalkan pengaturan GPU di Linux

1. Konfigurasikan pengaturan GPU agar menjadi persisten. Perintah ini memerlukan waktu beberapa menit untuk dijalankan.

   ```
   [ec2-user ~]$ sudo nvidia-persistenced
   ```

2. [Hanya instans G3] Nonaktifkan fitur autoboost untuk semua GPU pada instance.

   ```
   [ec2-user ~]$ sudo nvidia-smi --auto-boost-default=0
   ```

3. Atur semua kecepatan clock GPU ke frekuensi maksimumnya. Gunakan kecepatan clock memori dan grafis yang ditentukan dalam perintah berikut.

   Beberapa versi driver NVIDIA tidak mendukung pengaturan kecepatan clock aplikasi, dan menampilkan kesalahan `"Setting applications clocks is not supported for GPU..."`, yang bisa Anda abaikan.

   - Instans G3:

     ```
     [ec2-user ~]$ sudo nvidia-smi -ac 2505,1177
     ```

   - Instans G4dn:

     ```
     [ec2-user ~]$ sudo nvidia-smi -ac 5001,1590
     ```

   - Instans G5:

     ```
     [ec2-user ~]$ sudo nvidia-smi -ac 6250,1710
     ```

   - Instans G6, G6f, Gr6, dan Gr6f:

```
[ec2-user ~]$ sudo nvidia-smi -ac 6251,2040
```

- Instans G6e:

```
[ec2-user ~]$ sudo nvidia-smi -ac 9001,2520
```

- Instans G7e:

```
[ec2-user ~]$ sudo nvidia-smi -ac 12481,2430
```

- Instans G7:

```
[ec2-user ~]$ sudo nvidia-smi -ac 12501,2415
```

- Instans P3 dan P3dn:

```
[ec2-user ~]$ sudo nvidia-smi -ac 877,1530
```

- Instans P4d:

```
[ec2-user ~]$ sudo nvidia-smi -ac 1215,1410
```

- Instans P4de:

```
[ec2-user ~]$ sudo nvidia-smi -ac 1593,1410
```

- Instans P5:

```
[ec2-user ~]$ sudo nvidia-smi -ac 2619,1980
```

- Instans P5e dan P5en:

```
[ec2-user ~]$ sudo nvidia-smi -ac 3201,1980
```

- P6-B200 contoh:

```
[ec2-user ~]$ sudo nvidia-smi -ac 3996,1965
```

- P6-B300 contoh:

```
[ec2-user ~]$ sudo nvidia-smi -ac 3996,2032
```

Optimalkan pengaturan GPU di Windows

1. Buka PowerShell jendela dan arahkan ke folder instalasi NVIDIA.

   ```
   PS C:\> cd "C:\Windows\System32\DriverStore\FileRepository\nvgridsw_aws.inf_*\"
   ```

2. [Hanya instans G3] Nonaktifkan fitur autoboost untuk semua GPU pada instance.

   ```
   PS C:\> .\nvidia-smi --auto-boost-default=0
   ```

3. Atur semua kecepatan clock GPU ke frekuensi maksimumnya. Gunakan kecepatan clock memori dan grafis yang ditentukan dalam perintah berikut.

   Beberapa versi driver NVIDIA tidak mendukung pengaturan kecepatan clock aplikasi, dan menampilkan kesalahan `"Setting applications clocks is not supported for GPU..."`, yang bisa Anda abaikan.

   - Instans G3:

     ```
     PS C:\> .\nvidia-smi -ac "2505,1177"
     ```

   - Instans G4dn:

     ```
     PS C:\> .\nvidia-smi -ac "5001,1590"
     ```

   - Instans G5:

     ```
     PS C:\> .\nvidia-smi -ac "6250,1710"
     ```

   - Instans G6, G6f, Gr6, dan Gr6f:

     ```
     PS C:\> .\nvidia-smi -ac "6251,2040"
     ```

   - Instans G6e:

     ```
     PS C:\> .\nvidia-smi -ac "9001,2520"
     ```

- Instans P3 dan P3dn:

```
PS C:\> .\nvidia-smi -ac "877,1530"
```

## Mengatur tampilan 4K ganda pada instans G4ad Linux

Setelah meluncurkan instance G4ad, Anda dapat mengatur layar 4K ganda.

Untuk menginstal driver AMD dan mengkonfigurasi layar ganda

1. Hubungkan ke instans Linux Anda untuk mendapatkan alamat Bus PCI dari GPU yang ingin Anda targetkan untuk dual 4K (2x4k):

   ```
   lspci -vv | grep -i amd
   ```

   Output Anda serupa dengan berikut ini:

   ```
   00:1e.0 Display controller: Advanced Micro Devices, Inc. [*AMD*/ATI] Device 7362 (rev
    c3)
   Subsystem: Advanced Micro Devices, Inc. [AMD/ATI] Device 0a34
   ```

2. Harap perhatikan, alamat bus PCI adalah 00:1e.0 pada output di atas. Buat file bernama `/etc/modprobe.d/amdgpu.conf` dan tambahkan:

   ```
   options amdgpu virtual_display=0000:00:1e.0,2
   ```

3. Untuk menginstal driver AMD di Linux, lihatDriver AMD untuk instans EC2 Anda. Jika Anda sudah menginstal driver GPU AMD, Anda perlu membangun kembali modul kernel amdgpu melalui dkms.

4. Gunakan file xorg.conf di bawah ini untuk menentukan topologi layar ganda (2x4K) dan simpan file di `/etc/X11/xorg.conf`:

   ```
   ~$ cat /etc/X11/xorg.conf
   Section "ServerLayout"
       Identifier     "Layout0"
       Screen          0 "Screen0"
       Screen        1 "Screen1"
       InputDevice     "Keyboard0" "CoreKeyboard"
       InputDevice     "Mouse0" "CorePointer"
       Option          "Xinerama" "1"
   ```

```
EndSection
Section "Files"
    ModulePath "/opt/amdgpu/lib64/xorg/modules/drivers"
    ModulePath "/opt/amdgpu/lib/xorg/modules"
    ModulePath "/opt/amdgpu-pro/lib/xorg/modules/extensions"
    ModulePath "/opt/amdgpu-pro/lib64/xorg/modules/extensions"
    ModulePath "/usr/lib64/xorg/modules"
    ModulePath "/usr/lib/xorg/modules"
EndSection
Section "InputDevice"
    # generated from default
    Identifier     "Mouse0"
    Driver         "mouse"
    Option         "Protocol" "auto"
    Option         "Device" "/dev/psaux"
    Option         "Emulate3Buttons" "no"
    Option         "ZAxisMapping" "4 5"
EndSection
Section "InputDevice"
    # generated from default
    Identifier     "Keyboard0"
    Driver         "kbd"
EndSection

Section "Monitor"
    Identifier     "Virtual"
    VendorName     "Unknown"
    ModelName      "Unknown"
    Option         "Primary" "true"
EndSection

Section "Monitor"
    Identifier     "Virtual-1"
    VendorName     "Unknown"
    ModelName      "Unknown"
    Option         "RightOf" "Virtual"
EndSection

Section "Device"
    Identifier     "Device0"
    Driver         "amdgpu"
    VendorName     "AMD"
    BoardName      "Radeon MxGPU V520"
    BusID          "PCI:0:30:0"
```

```
EndSection

Section "Device"
    Identifier     "Device1"
    Driver         "amdgpu"
    VendorName     "AMD"
    BoardName      "Radeon MxGPU V520"
    BusID          "PCI:0:30:0"
EndSection

Section "Extensions"
    Option         "DPMS" "Disable"
EndSection

Section "Screen"
    Identifier     "Screen0"
    Device         "Device0"
    Monitor        "Virtual"
    DefaultDepth   24
    Option         "AllowEmptyInitialConfiguration" "True"
    SubSection "Display"
        Virtual    3840 2160
        Depth      32
    EndSubSection
EndSection

Section "Screen"
    Identifier     "Screen1"
    Device         "Device1"
    Monitor        "Virtual"
    DefaultDepth   24
    Option         "AllowEmptyInitialConfiguration" "True"
    SubSection "Display"
        Virtual    3840 2160
        Depth      32
    EndSubSection
EndSection
```

5. Siapkan DCV dengan mengikuti petunjuk dalam menyiapkan desktop interaktif.
6. Setelah pengaturan DCV selesai, lakukan reboot.
7. Pastikan driver berfungsi:

```
dmesg | grep amdgpu
```

Responsnya akan terlihat seperti berikut:

```
Initialized amdgpu
```

8. Anda akan melihat di output untuk `DISPLAY=:0 xrandr -q` bahwa Anda memiliki 2 tampilan virtual yang terhubung:

```
~$ DISPLAY=:0 xrandr -q
Screen 0: minimum 320 x 200, current 3840 x 1080, maximum 16384 x 16384
Virtual connected primary 1920x1080+0+0 (normal left inverted right x axis y axis)
 0mm x 0mm
 4096x3112  60.00
 3656x2664  59.99
 4096x2160  60.00
 3840x2160  60.00
 1920x1200  59.95
 1920x1080  60.00
 1600x1200  59.95
 1680x1050  60.00
 1400x1050  60.00
 1280x1024  59.95
 1440x900 59.99
 1280x960 59.99
 1280x854 59.95
 1280x800 59.96
 1280x720 59.97
 1152x768 59.95
 1024x768 60.00 59.95
 800x600  60.32 59.96 56.25
 848x480  60.00 59.94
 720x480  59.94
 640x480  59.94 59.94
Virtual-1 connected 1920x1080+1920+0 (normal left inverted right x axis y axis) 0mm x
 0mm
 4096x3112  60.00
 3656x2664  59.99
 4096x2160  60.00
 3840x2160  60.00
 1920x1200  59.95
```

```
 1920x1080  60.00
 1600x1200  59.95
 1680x1050  60.00
 1400x1050  60.00
 1280x1024  59.95
 1440x900 59.99
 1280x960 59.99
 1280x854 59.95
 1280x800 59.96
 1280x720 59.97
 1152x768 59.95
 1024x768 60.00 59.95
 800x600  60.32 59.96 56.25
 848x480  60.00 59.94
 720x480  59.94
640x480  59.94 59.94
```

9. Saat Anda terhubung ke DCV, ubah resolusi menjadi 2x4K, yang mengonfirmasi bahwa dukungan monitor ganda terdaftar oleh DCV.




## Siapkan desktop interaktif untuk Linux

Setelah Anda mengonfirmasi bahwa instans Linux Anda telah menginstal driver GPU AMD dan amdgpu sedang digunakan, Anda dapat menginstal manajer desktop interaktif. Kami merekomendasikan lingkungan desktop MATE untuk kompatibilitas dan performa terbaik.

### Prasyarat

Buka editor teks dan simpan yang berikut ini sebagai file bernama `xorg.conf`. Anda akan memerlukan file ini pada instans Anda.

```
Section "ServerLayout"
Identifier     "Layout0"
Screen          0 "Screen0"
InputDevice     "Keyboard0" "CoreKeyboard"
InputDevice     "Mouse0" "CorePointer"
EndSection
Section "Files"
ModulePath "/opt/amdgpu/lib64/xorg/modules/drivers"
```

```
ModulePath "/opt/amdgpu/lib/xorg/modules"
ModulePath "/opt/amdgpu-pro/lib/xorg/modules/extensions"
ModulePath "/opt/amdgpu-pro/lib64/xorg/modules/extensions"
ModulePath "/usr/lib64/xorg/modules"
ModulePath "/usr/lib/xorg/modules"
EndSection
Section "InputDevice"
# generated from default
Identifier     "Mouse0"
Driver         "mouse"
Option         "Protocol" "auto"
Option         "Device" "/dev/psaux"
Option         "Emulate3Buttons" "no"
Option         "ZAxisMapping" "4 5"
EndSection
Section "InputDevice"
# generated from default
Identifier     "Keyboard0"
Driver         "kbd"
EndSection
Section "Monitor"
Identifier     "Monitor0"
VendorName     "Unknown"
ModelName      "Unknown"
EndSection
Section "Device"
Identifier     "Device0"
Driver         "amdgpu"
VendorName     "AMD"
BoardName      "Radeon MxGPU V520"
BusID          "PCI:0:30:0"
EndSection
Section "Extensions"
Option         "DPMS" "Disable"
EndSection
Section "Screen"
Identifier     "Screen0"
Device         "Device0"
Monitor        "Monitor0"
DefaultDepth   24
Option         "AllowEmptyInitialConfiguration" "True"
SubSection "Display"
    Virtual    3840 2160
    Depth      32
```

```
EndSubSection
EndSection
```

Untuk menyiapkan desktop interaktif di Amazon Linux 2

1. Instal repositori EPEL.

   ```
   [ec2-user ~]$ sudo amazon-linux-extras install epel -y
   ```

2. Instal desktop MATE.

   ```
   [ec2-user ~]$ sudo amazon-linux-extras install mate-desktop1.x -y
   [ec2-user ~]$ sudo yum groupinstall "MATE Desktop" -y
   [ec2-user ~]$ sudo systemctl disable firewalld
   ```

3. Salin file `xorg.conf` ke `/etc/X11/xorg.conf`.
4. Boot ulang instans.

   ```
   [ec2-user ~]$ sudo reboot
   ```

5. (Opsional) Instal server Amazon DCV untuk menggunakan Amazon DCV sebagai protokol tampilan berkinerja tinggi, lalu sambung ke sesi Amazon DCV menggunakan klien pilihan Anda.

Untuk menyiapkan desktop interaktif di Ubuntu

1. Instal desktop MATE.

   ```
   $ sudo apt install xorg-dev ubuntu-mate-desktop -y
   $ sudo apt purge ifupdown -y
   ```

2. Salin file `xorg.conf` ke `/etc/X11/xorg.conf`.
3. Boot ulang instans.

   ```
   $ sudo reboot
   ```

4. Instal enkoder AMF untuk versi Ubuntu yang sesuai.

   ```
   $ sudo apt install ./amdgpu-pro-20.20-*/amf-amdgpu-pro_20.20-*_amd64.deb
   ```

5. (Opsional) Instal server Amazon DCV untuk menggunakan Amazon DCV sebagai protokol tampilan berkinerja tinggi, lalu sambungkan ke sesi Amazon DCV menggunakan klien pilihan Anda.
6. Setelah penginstalan DCV memberikan izin video Pengguna DCV:

```
$ sudo usermod -aG video dcv
```

Untuk menyiapkan desktop interaktif di CentOS

1. Instal repositori EPEL.

```
$ sudo yum update -y
$ sudo yum install epel-release -y
```

2. Instal desktop MATE.

```
$ sudo yum groupinstall "MATE Desktop" -y
$ sudo systemctl disable firewalld
```

3. Salin file `xorg.conf` ke `/etc/X11/xorg.conf`.
4. Boot ulang instans.

```
$ sudo reboot
```

5. (Opsional) Instal server Amazon DCV untuk menggunakan Amazon DCV sebagai protokol tampilan berkinerja tinggi, lalu sambungkan ke sesi Amazon DCV menggunakan klien pilihan Anda.

## Memulai dengan instans yang dipercepat GPU

Jenis instans akselerasi GPU generasi terbaru, seperti yang ditunjukkan dalam daftar berikut memberikan kemampuan kinerja tertinggi untuk aplikasi pembelajaran mendalam dan komputasi kinerja tinggi (HPC). Pilih tautan jenis instans untuk mempelajari lebih lanjut tentang kemampuannya.

- Keluarga P6
- Keluarga P6
- Keluarga P5

Untuk daftar lengkap spesifikasi tipe instans untuk jenis instans yang dipercepat, lihat Komputasi yang dipercepat di referensi Jenis Instans Amazon EC2.

Konfigurasi perangkat lunak

Cara termudah untuk memulai dengan jenis instans akselerasi GPU generasi terbaru adalah dengan meluncurkan instance dari AMI Pembelajaran AWS Mendalam yang telah dikonfigurasi sebelumnya dengan semua perangkat lunak yang diperlukan. Untuk informasi terbaru AWS Deep Learning AMIs untuk digunakan dengan jenis instans akselerasi GPU, lihat DLAMI yang Didukung P6 di Panduan Peng AWS Deep Learning AMIs embang.

Jika Anda perlu membuat AMI khusus untuk meluncurkan instans yang menghosting aplikasi pembelajaran mendalam atau HPC, sebaiknya instal versi perangkat lunak minimum berikut di atas image dasar Anda.

| Tipe instans | Driver NVIDIA | CUDA | NVIDIA GDRCopy | Penginstal EFA | NCCL | EFA K8s [1] |
|---|---|---|---|---|---|---|
| G7 | 595 | 13.2 | 2.5.1 | 1.48.0 | 2.29.7-1 | 0.5.10 |
| G7e | 575 | 12.9 | 2.5 | 1.45.0 | 2.28.3 | 0.5.10 |
| P5 | 530 | 12.1 | 2.3 | 1.24.1 | 2.18.3 | 0.4.4 |
| P5.4xlarge | 530 | 12.1 | 2.3 | 1,43,1[2] | 2.18.3 | 0.4.4 |
| P5e | 550 | 12.1 | 2.3 | 1.24.1 | 2.18.3 | 0.5.5 |
| P5en | 550 | 12.1 | 2.3 | 1.24.1 | 2.18.3 | 0.5.6 |
| P6-B200 | 570 | 12.8 | 2.5 | 1.41.0 | 2.26.21 | 0.5.10 |
| P6e-GB200 | 570 | 12.8 | 2.5 | 1.41.0 | 2.26.21 | 0.5.10 |
| P6-B300 | 580 | 13.0 | 2.5 | 1.44.0 | 2.28.3 | 0.5.10 |

[1] Kol om EFA K8s berisi versi minimum yang direkomendasikan untuk. `aws-efa-k8s-device-plugin`

[2] Ada masalah kompatibilitas yang mempengaruhi `P5.4xlarge` contoh ketika GPU-to-GPU komunikasi menggunakan Elastic Fabric Adapter (EFA) dan NVIDIA Collective Communications Library (NCCL). Untuk mengurangi masalah, atur variabel lingkungan `FI_HMEM_DISABLE_P2P` ke`1`, dan pastikan Anda menginstal EFA versi 1.43.1 atau yang lebih baru.

> **Note**
>
> Jika Anda menggunakan versi 1.41.0 dari penginstal EFA, diser `aws-ofi-nccl plugin` takan dengan itu. Untuk versi penginstal EFA yang lebih lama, gunakan `aws-ofi-nccl plugin` versi `1.7.2-aws` atau yang lebih baru.

Kami juga menyarankan Anda mengonfigurasi instance agar tidak menggunakan lebih dalam C-states. Untuk informasi selengkapnya, lihat Kinerja tinggi dan latensi rendah dengan membatasi lebih C-states dalam di Panduan Pengguna Amazon Linux 2. AMI GPU AWS Deep Learning Base terbaru telah dikonfigurasi sebelumnya untuk tidak menggunakan lebih dalam. C-states

Untuk konfigurasi jaringan dan Elastic Fabric Adapter (EFA) lihatMaksimalkan bandwidth jaringan pada instans Amazon EC2 dengan beberapa kartu jaringan.

## Instans Mac Amazon EC2

Instans Mac EC2 ideal untuk mengembangkan, membangun, menguji, dan menandatangani aplikasi untuk platform Apple, seperti iPhone, iPad, Mac, Vision Pro, Apple Watch, Apple TV, dan Safari. Anda dapat terhubung ke instans Mac Anda menggunakan SSH atau Apple Remote Desktop (ARD).

> **Note**
>
> Unit penagihan adalah host khusus. Instans yang berjalan pada host itu tidak dikenai biaya tambahan.

Instans Amazon EC2 Mac secara native mendukung sistem operasi macOS.

- Instans Mac EC2 x86 (`mac1.metal`) dibangun di atas perangkat keras Mac mini 2018 yang didukung oleh prosesor Intel generasi kedelapan (Coffee Lake) 3.2 GHz Core i7, 6 inti fisik dan 12 inti logis, dan memori 32 GiB.

- Instans Mac EC2 M1 (`mac2.metal`) dibangun di atas perangkat keras mini Mac 2020 yang didukung oleh prosesor Apple silikon M1, 8 inti CPU, 8 inti GPU, memori 16 GiB, dan Mesin Neural Apple 16-inti.
- Instans EC2 M1 Ultra Mac (`mac2-m1ultra.metal`) dibangun di atas perangkat keras Mac Studio 2022 yang didukung oleh prosesor Apple silikon M1 Ultra, 20 inti CPU, 64 inti GPU, memori 128 GiB, dan Apple Neural Engine 32-core.
- Instans Mac EC2 M2 (`mac2-m2.metal`) dibangun di atas perangkat keras mini Mac 2023 yang didukung oleh prosesor Apple silicon M2, 8 inti CPU, 10 inti GPU, memori 24 GiB, dan Mesin Neural Apple 16-inti.
- Instans EC2 M2 Pro Mac (`mac2-m2pro.metal`) dibangun di atas perangkat keras mini Mac 2023 yang didukung oleh prosesor Apple silikon M2 Pro, 12 inti CPU, 19 inti GPU, memori 32 GiB, dan 16-core Apple Neural Engine.
- Instans Mac EC2 M4 (`mac-m4.metal`) dibangun di atas perangkat keras mini Mac 2024 yang didukung oleh prosesor Apple silicon M4, 10 inti CPU, 10 inti GPU, memori 24 GiB, dan 16-core Apple Neural Engine.
- Instans Mac EC2 M4 Pro Mac (`mac-m4pro.metal`) dibangun di atas perangkat keras mini Mac 2024 yang didukung oleh prosesor Apple silicon M4 Pro, 14 inti CPU, 20 inti GPU, memori 48 GiB, dan 16-core Apple Neural Engine.

Amazon EC2 Mac Dedicated Host mendukung pemulihan otomatis Host Khusus dan pemeliharaan host berbasis reboot.

Daftar Isi


- Pertimbangan-pertimbangan
- Kesiapan instans
- AMI macOS EC2
- EC2 macOS Init
- Monitor Sistem Amazon EC2 untuk macOS
- Sumber daya terkait
- Luncurkan instans Mac menggunakan konsol Konsol Manajemen AWS atau AWS CLI
- Sambungkan ke instance Mac Anda menggunakan SSH atau GUI
- Memperbarui sistem operasi dan perangkat lunak pada instans Amazon EC2 Mac
- Meningkatkan ukuran volume EBS di instans Mac Anda

- Menghentikan atau menghentikan instans Mac Amazon EC2 Anda
- Konfigurasikan Perlindungan Integritas Sistem untuk instans Mac Amazon EC2
- Temukan versi macOS yang didukung untuk Amazon EC2 Mac Dedicated Host
- Berlangganan notifikasi AMI macOS
- Ambil ID AMI macOS menggunakan AWS Systems Manager API Penyimpanan Parameter
- Catatan rilis AMI macOS Amazon EC2

## Pertimbangan-pertimbangan

Pertimbangan berikut berlaku untuk instans Mac:

- Instans Mac hanya tersedia sebagai instans bare metal di Host Khusus, dengan masa alokasi minimal 24 jam sebelum Anda dapat membagikan Host Khusus. Anda dapat meluncurkan satu instans Mac per Host Khusus. Anda dapat berbagi Host Khusus dengan AWS akun atau unit organisasi dalam AWS organisasi Anda, atau seluruh AWS organisasi.
- Instans Mac tersedia dalam berbagai versi Wilayah AWS. Untuk daftar ketersediaan instans Mac di masing-masing instans Wilayah AWS, lihat Jenis instans Amazon EC2 berdasarkan Wilayah.
- Instans Mac hanya tersedia sebagai On-Demand Instans. Instans tersebut tidak tersedia sebagai Instans Spot atau Instans Terpesan. Anda dapat menghemat uang pada instans Mac dengan membeli Savings Plans.
- Kompatibilitas berbagai jenis instans Mac dengan Amazon Machine Images (AMI) macOS tertentu bervariasi. Untuk informasi selengkapnya, lihat Catatan rilis AMI macOS Amazon EC2.
- EBSHotplug didukung.
- AWS tidak mengelola atau mendukung SSD internal pada perangkat keras Apple. Kami sangat menyarankan Anda menggunakan volume Amazon EBS sebagai gantinya. EBSvolume memberikan manfaat elastisitas, ketersediaan, dan daya tahan yang sama pada instans Mac seperti pada instans EC2 lainnya.
- Sebaiknya gunakan volume Amazon EBS dengan 10.000 IOPS dan 400 MiB/s throughput dengan instans Mac untuk kinerja optimal. Untuk informasi selengkapnya, lihat Jenis volume Amazon EBS di Panduan Pengguna Amazon EBS.
- Instans Mac mendukung Amazon EC2 Auto Scaling.
- Pada instans x86 Mac, pembaruan perangkat lunak otomatis dinonaktifkan. Kami merekomendasikan Anda untuk menerapkan pembaruan dan mengujinya pada instans Anda

sebelum memasukkan instans ke dalam proses produksi. Untuk informasi selengkapnya, lihat Memperbarui sistem operasi dan perangkat lunak pada instans Amazon EC2 Mac.

- Ketika Anda menghentikan atau mengakhiri instans Mac, alur kerja scrubbing dilakukan pada Host Khusus. Untuk informasi selengkapnya, lihat Menghentikan atau menghentikan instans Mac Amazon EC2 Anda.
- **Important**

  Fitur Apple Intelligence tidak tersedia saat mem-boot perangkat keras Mac dari volume eksternal. Karena instans EC2 Mac boot dari volume EBS eksternal secara default, instans tersebut tidak mendukung fitur Apple Intelligence.
- **Warning**

  Jangan gunakan FileVault. Jika Anda mengaktifkan FileVault, host gagal melakukan boot karena partisi terkunci. Jika enkripsi data diperlukan, gunakan enkripsi Amazon EBS untuk menghindari masalah boot dan dampak kinerja. Dengan enkripsi Amazon EBS, operasi enkripsi terjadi di server host, memastikan keamanan data-at-rest dan data-in-transit antara instans dan penyimpanan EBS terlampir. Untuk informasi selengkapnya, lihat enkripsi Amazon EBS di Panduan Pengguna Amazon EBS.

## Kesiapan instans

Setelah meluncurkan instans Mac, Anda harus menunggu hingga instans siap sebelum dapat terhubungkan ke instans tersebut. Untuk AMI AWS yang dijual dengan instance Mac x86 atau instance Mac silikon Apple, waktu peluncuran dapat berkisar dari sekitar 6 menit hingga 20 menit. Bergantung pada ukuran volume Amazon EBS yang dipilih, penyertaan skrip tambahan ke data pengguna, atau perangkat lunak tambahan yang dimuat pada macOS AMI kustom, waktu peluncuran mungkin meningkat.

Anda dapat menggunakan skrip shell kecil, seperti yang di bawah ini, untuk melakukan polling API describe-instance-status untuk mengetahui kapan instans siap untuk dihubungkan. Dalam perintah berikut, ganti ID instans contoh dengan milik Anda.

```
for i in $(seq 1 200); do aws ec2 describe-instance-status --instance-
ids=i-1234567890abcdef0 \
    --query='InstanceStatuses[0].InstanceStatus.Status'; sleep 5; done;
```

## AMI macOS EC2

Amazon EC2 macOS dirancang untuk menyediakan lingkungan yang stabil, aman, dan mempunyai performa tinggi untuk beban kerja developer yang berjalan pada instans Mac Amazon EC2. AMI macOS EC2 menyertakan paket yang memungkinkan integrasi dengan mudah AWS, seperti alat konfigurasi peluncuran dan AWS perpustakaan dan alat populer.

Untuk informasi selengkapnya tentang AMI EC2 macOS, lihatCatatan rilis AMI macOS Amazon EC2.

AWS menyediakan AMI EC2 macOS yang diperbarui secara teratur yang mencakup pembaruan untuk paket yang dimiliki oleh AWS dan versi macOS terbaru yang telah teruji sepenuhnya. Selain itu, AWS menyediakan AMI yang telah diperbarui dengan pembaruan versi minor terbaru atau pembaruan versi mayor segera setelah sepenuhnya diuji dan diperiksa. Jika Anda tidak perlu menyimpan data atau melakukan penyesuaian pada instans Mac, Anda bisa mendapatkan pembaruan terkini dengan meluncurkan instans baru menggunakan AMI saat ini dan kemudian mengakhiri instans sebelumnya. Atau, Anda dapat memilih pembaruan mana yang akan diterapkan pada instans Mac Anda.

Untuk informasi tentang cara berlangganan notifikasi AMI macOS, lihatBerlangganan notifikasi AMI macOS.

## EC2 macOS Init

EC2macOS Init digunakan untuk menginisialisasi instance EC2 Mac saat peluncuran. Init ini menggunakan grup prioritas untuk menjalankan grup logis tugas pada saat bersamaan.

File plist yang diluncurkan adalah `/Library/LaunchDaemons/com.amazon.ec2.macos-init.plist`. File untuk Init macOS EC2 terletak di `/usr/local/aws/ec2-macos-init`.

Untuk informasi selengkapnya, lihat https://github.com/aws/ec2-macos-init.

## Monitor Sistem Amazon EC2 untuk macOS

Amazon EC2 System Monitor untuk macOS menyediakan metrik pemanfaatan CPU ke Amazon CloudWatch. Ini mengirimkan metrik ini CloudWatch ke perangkat serial khusus dalam periode 1 menit. Anda dapat mengaktifkan atau menonaktifkan agen ini sebagai berikut. Agen tidak diaktifkan secara default.

```
sudo setup-ec2monitoring [enable | disable]
```

> **Note**
>
> Amazon EC2 System Monitor untuk macOS saat ini tidak didukung pada instans Mac silikon Apple.

## Sumber daya terkait

Untuk informasi tentang harga, lihat Harga.

Untuk informasi selengkapnya tentang instans Mac, lihat Instans Mac Amazon EC2.

Untuk informasi selengkapnya tentang spesifikasi perangkat keras dan performa jaringan instans Mac, lihat Instans tujuan umum.

## Luncurkan instans Mac menggunakan konsol Konsol Manajemen AWS atau AWS CLI

Instans EC2 Mac memerlukan Host Khusus. Pertama-tama, Anda harus mengalokasikan host ke akun Anda, kemudian meluncurkan instans ke host.

Anda dapat meluncurkan instance Mac menggunakan Konsol Manajemen AWS atau AWS CLI.

Luncurkan instans Mac menggunakan konsol

Untuk meluncurkan instans Mac ke Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Alokasikan Dedicated Host, sebagai berikut:
    a. Di panel navigasi, pilih Host Khusus.
    b. Pilih Alokasikan Host Khusus, lalu lakukan hal berikut:
        i. Untuk keluarga Inst ance, pilih keluarga Instance Mac. Jika keluarga instans tidak muncul dalam daftar, berarti tidak didukung di Wilayah yang saat ini dipilih.
        ii. Untuk Jenis Inst ance, pilih jenis instance berdasarkan keluarga instans yang dipilih yang dipilih.
        iii. Untuk Zona Ketersediaan, pilih Zona Ketersediaan untuk Host Khusus.
        iv. Untuk Jumlah, biarkan di 1.
        v. Pilih Alokasikan.

3. Luncurkan instans pada host, sebagai berikut:

   a. Pilih Host Khusus yang Anda buat, lalu lakukan hal berikut:

      i. Pilih Tindakan, Luncurkan instans ke host.

      ii. Pada Aplikasi dan Gambar OS (Amazon Machine Image), pilih AMI macOS.

      iii. Di bawah Jenis instans, pilih jenis instance Mac.

      iv. Pada Detail lanjutan, pastikan bahwa Penghunian, Host penghunian oleh, dan ID host penghunian telah dikonfigurasi sebelumnya berdasarkan Host Khusus yang Anda buat. Perbarui Afinitas penghunian sesuai kebutuhan.

      v. Selesaikan wizard, tentukan volume EBS, grup keamanan, dan pasangan kunci yang diperlukan.

      vi. Di panel Ringkasan, pilih Luncurkan instans.

   b. Halaman konfirmasi memberi tahu Anda bahwa instans Anda akan diluncurkan. Pilih Lihat semua instans untuk menutup halaman konfirmasi dan kembali ke konsol. Status awal dari sebuah instans adalah `pending`. Instans siap ketika statusnya berubah menjadi `running` dan lolos pemeriksaan status.

Luncurkan instans Mac menggunakan konsol AWS CLI

Mengalokasikan Host Khusus

Gunakan perintah allocate-hosts berikut untuk mengalokasikan Host Khusus untuk instance Mac Anda, menggantinya `instance-type` dengan jenis instance mac yang valid, dan `region` dan `availability-zone` dengan yang sesuai untuk lingkungan Anda.

```
aws ec2 allocate-hosts --region us-east-1 --instance-type mac1.metal --availability-
zone us-east-1b --auto-placement "on" --quantity 1
```

Luncurkan instans di host

Gunakan perintah run-instance berikut untuk meluncurkan instance Mac, sekali lagi mengganti `instance-type` dengan jenis instance mac yang valid, dan `region` dan `availability-zone` dengan yang digunakan sebelumnya.

```
aws ec2 run-instances --region us-east-1 --instance-type mac1.metal --placement
 Tenancy=host --image-id ami_id --key-name my-key-pair
```

Status awal dari sebuah instans adalah `pending`. Instans siap ketika statusnya berubah menjadi `running` dan lolos pemeriksaan status. Gunakan perintah describe-instance-status berikut untuk menampilkan informasi status untuk instans Anda.

```
aws ec2 describe-instance-status --instance-ids i-017f8354e2dc69c4f
```

Berikut ini adalah contoh output untuk sebuah instans yang sedang berjalan dan telah lulus pemeriksaan status.


```
{
    "InstanceStatuses": [
        {
            "AvailabilityZone": "us-east-1b",
            "InstanceId": "i-017f8354e2dc69c4f",
            "InstanceState": {
                "Code": 16,
                "Name": "running"
            },
            "InstanceStatus": {
                "Details": [
                    {
                        "Name": "reachability",
                        "Status": "passed"
                    }
                ],
                "Status": "ok"
            },
            "SystemStatus": {
                "Details": [
                    {
                        "Name": "reachability",
                        "Status": "passed"
                    }
                ],
                "Status": "ok"
            }
        }
    ]
}
```

## Sambungkan ke instance Mac Anda menggunakan SSH atau GUI

Anda dapat terhubung ke instance Mac menggunakan SSH atau antarmuka pengguna grafis (GUI).

Beberapa pengguna dapat mengakses OS secara bersamaan. Biasanya ada sesi 1:1 User:GUI, karena layanan Berbagi Layar bawaan pada port 5900. Menggunakan SSH dalam macOS mendukung beberapa sesi hingga batas “Sesi Maks” dalam `sshd_config` file.

Terhubung ke instans Anda menggunakan SSH

Instans Mac Amazon EC2 tidak mengizinkan SSH root jarak jauh secara default. ec2-userAkun dikonfigurasi untuk masuk dari jarak jauh menggunakan SSH. Akun ec2-user ini juga memiliki sudo hak istimewa. Setelah Anda terhubung ke instans, Anda dapat menambahkan pengguna lain.

Untuk mendukung koneksi ke instans Anda menggunakan SSH, luncurkan instans menggunakan pasangan kunci dan grup keamanan yang mengizinkan akses SSH, dan pastikan bahwa instans tersebut memiliki konektivitas internet. Anda menyediakan file `.pem` untuk pasangan kunci saat Anda terhubung ke instans.

Gunakan prosedur berikut untuk menghubungkan ke instans Mac Anda menggunakan klien SSH. Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux.

Untuk terhubung ke instans Anda menggunakan SSH

1. Pastikan bahwa komputer lokal Anda memiliki klien SSH yang terinstal dengan memasukkan ssh di baris perintah. Jika komputer Anda tidak mengenali perintah tersebut, cari klien SSH untuk sistem operasi Anda lalu instal.
2. Dapatkan nama DNS publik dari instans Anda. Dengan menggunakan konsol Amazon EC2, Anda dapat menemukan nama DNS publik di tab Detail dan tab Jaringan. Dengan menggunakan AWS CLI, Anda dapat menemukan nama DNS publik menggunakan perintah describe-instance.
3. Temukan file `.pem` untuk pasangan kunci yang Anda tentukan saat meluncurkan instans.
4. Hubungkan ke instans Anda menggunakan perintah ssh berikut, yang menentukan nama DNS publik dari instans dan file `.pem`.

```
ssh -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

Autentikasi kata sandi dinonaktifkan untuk mencegah serangan kata sandi brute-force. Sebelum Anda membuat perubahan pada konfigurasi SSH, buka `/usr/local/aws/ec2-macos-init/init.toml` dan atur `secureSSHDConfig` ke`false`.

Hubungkan ke antarmuka pengguna grafis (GUI) instans Anda

Gunakan prosedur berikut untuk terhubung ke GUI instans Anda menggunakan VNC, Apple Remote Desktop (ARD), atau aplikasi Apple Screen Sharing (disertakan dengan macOS).

> **Note**
>
> macOS 10.14 dan setelahnya hanya mengizinkan kontrol jika Berbagi Layar diaktifkan melalui Preferensi Sistem.

Untuk terhubung ke instans Anda menggunakan klien ARD atau klien VNC

1. Pastikan komputer lokal Anda menginstal klien ARD atau klien VNC yang mendukung ARD. Pada macOS, Anda dapat memanfaatkan aplikasi Berbagi Layar bawaan. Jika tidak, cari ARD untuk sistem operasi Anda dan instal.
2. Dari komputer lokal Anda, hubungkan ke instans Anda menggunakan SSH.
3. Siapkan kata sandi untuk akun ec2-user menggunakan perintah passwd sebagai berikut.

   ```
   [ec2-user ~]$ sudo passwd ec2-user
   ```

4. Instal dan mulai macOS Screen Sharing menggunakan perintah berikut.

   ```
   [ec2-user ~]$ sudo launchctl enable system/com.apple.screensharing
   sudo launchctl load -w /System/Library/LaunchDaemons/com.apple.screensharing.plist
   ```

5. Putuskan sambungan dari instans dengan mengetik exit dan menekan Enter.
6. Dari komputer Anda, hubungkan ke instans Anda menggunakan perintah ssh berikut ini. Selain opsi yang ditunjukkan di bagian sebelumnya, gunakan opsi -L untuk mengaktifkan penerusan port dan meneruskan semua lalu lintas di port lokal 5900 ke server ARD pada instans.

   ```
   ssh -L 5900:localhost:5900 -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
   ```

7. Dari komputer lokal Anda, gunakan klien ARD atau klien VNC yang mendukung ARD untuk terhubung ke `localhost:5900`. Misalnya, gunakan aplikasi Berbagi Layar pada macOS sebagai berikut:

   a. Buka Finder dan pilih Go.

   b. Pilih Hubungkan ke Server.

   c. Di bidang Alamat Server, masukkan `vnc://localhost:5900`.

   d. Masuk seperti yang diminta, gunakan **`ec2-user`** sebagai nama pengguna dan kata sandi yang Anda buat untuk akun pengguna ec2.

Mengubah resolusi layar macOS di instans Mac

Setelah tersambung ke instans Mac EC2 menggunakan klien ARD atau klien VNC yang mendukung ARD, Anda dapat mengubah resolusi layar lingkungan macOS menggunakan alat atau utilitas macOS yang tersedia untuk umum, seperti displayplacer.

Untuk mengubah resolusi layar menggunakan displayplacer

1. Instal displayplacer.

```
[ec2-user ~]$ brew tap jakehilborn/jakehilborn && brew install displayplacer
```

2. Tampilkan informasi layar saat ini dan kemungkinan resolusi layar.

```
[ec2-user ~]$ displayplacer list
```

3. Terapkan resolusi layar yang diinginkan.

```
[ec2-user ~]$ displayplacer "id:<screenID> res:<width>x<height> origin:(0,0)
 degree:0"
```

   Contoh:

```
RES="2560x1600"
displayplacer "id:69784AF1-CD7D-B79B-E5D4-60D937407F68 res:${RES} scaling:off
 origin:(0,0) degree:0"
```

# Memperbarui sistem operasi dan perangkat lunak pada instans Amazon EC2 Mac

Topik berikut menjelaskan cara memperbarui sistem operasi dan perangkat lunak pada instans Mac silikon Apple (Mac2,,, Mac2-m1ultra, Mac2-m2 Mac2-m2pro Mac-m4, dan Mac-m4pro) dan instans Mac x86 (Mac1).

**Warning**

Pemasangan versi beta atau pratinjau macOS hanya tersedia pada instans Mac silikon Apple. Amazon EC2 tidak memenuhi syarat versi beta atau pratinjau macOS dan tidak memastikan instans akan tetap berfungsi setelah pembaruan ke versi macOS praproduksi. Mencoba menginstal versi beta atau pratinjau macOS di instans Amazon EC2 x86 Mac akan menyebabkan degradasi Host Khusus Amazon EC2 Mac saat Anda menghentikan atau mengakhiri instans, dan akan menghambat Anda untuk memulai atau meluncurkan instans baru di host tersebut.

**Note**

Jika Anda melakukan pembaruan macOS di tempat sebelum AWS merilis AMI resmi, pembaruan hanya berlaku untuk host yang dipilih. Jika Anda memiliki host lain, atau jika Anda meluncurkan host baru, Anda harus melakukan proses pembaruan yang sama pada host tersebut juga. Setiap versi macOS memerlukan versi firmware minimum pada perangkat keras Apple Mac yang mendasarinya. Pembaruan di tempat hanya memperbarui firmware pada host yang dipilih dan tidak mentransfer ke host lain yang ada atau baru. Untuk memeriksa versi macOS mana yang kompatibel dengan Host Khusus Amazon EC2 Mac Anda, lihat Temukan versi macOS yang didukung untuk Host Khusus Amazon EC2 Mac Anda.

## Perbarui perangkat lunak pada instans Mac Apple silicon

### Prasyarat

Karena pembaruan dalam konfigurasi driver jaringan, driver ENA versi 1.0.2 tidak kompatibel dengan macOS 13.3 dan yang lebih baru. Jika Anda ingin menginstal versi beta, pratinjau, atau produksi macOS versi 13.3 atau setelahnya dan belum menginstal driver ENA terbaru, gunakan prosedur berikut untuk menginstal versi driver yang baru.

Untuk menginstal driver ENA versi baru

1. Di jendela Terminal, sambungkan ke instans Mac Apple silicon menggunakan SSH.
2. Perbarui Homebrew dan unduh aplikasi ENA ke dalam `Applications` file menggunakan perintah berikut.

```
[ec2-user ~]$ brew update
```

```
[ec2-user ~]$ brew install amazon-ena-ethernet-dext
```

3. Putuskan sambungan dari instans Anda dengan mengetik exit dan menekan kembali.
4. Gunakan klien VNC untuk mengaktifkan aplikasi ENA.
   a. Siapkan klien VNC menggunakan Hubungkan ke antarmuka pengguna grafis (GUI) instans Anda.
   b. Setelah Anda terhubung ke instance Anda menggunakan aplikasi Berbagi Layar, buka folder  Aplikasi dan buka aplikasi ENA.
   c. Pilih Aktifkan
   d. Untuk mengonfirmasi bahwa driver diaktifkan dengan benar, jalankan perintah berikut di jendela Terminal. Output dari perintah menunjukkan bahwa driver lama dalam keadaan diakhiri dan driver baru dalam keadaan diaktifkan.

```
systemextensionsctl list;
```

   e. Setelah Anda memulai ulang instans, hanya driver baru yang akan hadir.

Lakukan pembaruan perangkat lunak

Pada instans Mac Apple silicon, Anda harus menyelesaikan beberapa langkah untuk melakukan pembaruan sistem operasi di tempat. Ini termasuk mendelegasikan kepemilikan volume root Amazon EBS ke pengguna administratif volume root EBS. Anda dapat memilih untuk melakukan ini secara otomatis menggunakan Amazon EC2 API, atau Anda dapat melakukannya secara manual dengan menjalankan perintah pada instans Anda.

Automated volume ownership delegation (Recommended)

Pertimbangan-pertimbangan

- Diperlukan waktu antara 30 dan 90 menit untuk menyelesaikan tugas delegasi kepemilikan volume. Selama waktu ini, instance tidak dapat dijangkau.
- Versi macOS berikut didukung:
  - Mac2 | Mac2-m1ultra - macOS Ventura (versi 13.0 atau lebih baru)
  - Mac2-m2 | Mac2-m2pro - macOS Ventura (versi 13.2 atau lebih baru)
  - Mac-m4 | Mac-m4pro - macOS Sequoia (versi 15.6 atau lebih baru)
- Instans hanya harus memiliki satu volume yang dapat di-boot, dan setiap volume terlampir hanya dapat memiliki satu pengguna admin tambahan.

Langkah 1: Tetapkan kata sandi dan aktifkan token aman untuk pengguna administratif volume root EBS

Anda harus menetapkan kata sandi dan mengaktifkan token aman untuk pengguna administratif volume root Amazon EBS (`ec2-user`).

> **Note**
>
> Kata sandi dan token aman diatur saat pertama kali Anda terhubung ke instance Mac silikon Apple menggunakan GUI. Jika sebelumnya Anda terhubung ke instance menggunakan GUI, Anda tidak perlu melakukan langkah-langkah ini.

1. Hubungkan ke instance menggunakan SSH.
2. Tetapkan kata sandi untuk `ec2-user` pengguna.

```
$ sudo /usr/bin/dscl . -passwd /Users/ec2-user
```

3. Aktifkan token aman untuk `ec2-user` pengguna. Untuk `-oldPassword`, tentukan kata sandi yang sama dari langkah sebelumnya. Untuk `-newPassword`, tentukan kata sandi yang berbeda. Perintah berikut mengasumsikan bahwa Anda memiliki kata sandi lama dan baru yang disimpan dalam `.txt` file.

```
$ sysadminctl -oldPassword `cat old_password.txt` -newPassword `cat
 new_password.txt`
```

4. Pastikan token aman diaktifkan.

```
$ sysadminctl -secureTokenStatus ec2-user
```

Langkah 2: Mendelegasikan kepemilikan volume root Amazon EBS ke pengguna administratif volume root EBS

Untuk mendelegasikan kepemilikan, Anda harus membuat tugas delegasi kepemilikan volume.

1. Gunakan perintah create-delegate-mac-volum e-ownership-task untuk membuat tugas. Untuk`--instance-id`, tentukan ID instance. Untuk`--mac-credentials`, tentukan kredenSIAL berikut:

   - Pengguna administratif disk internal
     - Nama pengguna — Hanya pengguna administratif default (`aws-managed-user`) yang didukung dan digunakan secara default. Anda tidak dapat menentukan pengguna administratif yang berbeda.
     - Kata Sandi — Jika Anda tidak mengubah kata sandi default untuk`aws-managed-user`, tentukan kata sandi default, yang kosong. Jika tidak, tentukan kata sandi Anda.
   - Pengguna administratif volume root Amazon EBS
     - Nama Pengguna — Jika Anda tidak mengubah pengguna administratif default, tentukan`ec2-user`. Jika tidak, tentukan nama pengguna untuk pengguna administratif Anda.
     - Kata Sandi — Tentukan kata sandi yang Anda tetapkan untuk pengguna admin volume root pada Langkah 1 di atas.

   ```
   aws ec2 create-delegate-mac-volume-ownership-task \
   --instance-id i-1234567890abcdef0 \
   --mac-credentials file://mac-credentials.json
   ```

   Berikut ini adalah isi `mac-credentials.json` file yang direferensikan dalam contoh sebelumnya.

```
{
  "internalDiskPassword":"internal-disk-admin_password",
  "rootVolumeUsername":"root-volume-admin_username",
  "rootVolumePassword":"root-volume-admin_password"
}
```

2. Tunggu tugas delegasi kepemilikan volume selesai dan instans kembali ke keadaan sehat. Gunakan perintah describe-mac-modifiation-tasks. Untuk`--mac-modification-task-id`, tentukan ID tugas delegasi kepemilikan volume dari langkah sebelumnya.

```
aws ec2 describe-mac-modification-tasks \
--mac-modification-task-id task-id
```

3. Setelah tugas delegasi kepemilikan volume selesai, lanjutkan ke Langkah 3.

Langkah 3: Perbarui perangkat lunak

Setelah Anda mendelegasikan kepemilikan volume root Amazon EBS, ikuti langkah-langkah yang dijelaskan di Perbarui perangkat lunak pada instans Mac x86 (di bawah) untuk memperbarui perangkat lunak.

Manual volume ownership delegation

Saat Anda mengerjakan prosedur ini, Anda membuat dua kata sandi. Satu kata sandi adalah untuk pengguna administratif volume root Amazon EBS (`ec2-user`), dan kata sandi lainnya untuk pengguna administratif disk internal (`aws-managed-user`). Ingat kata sandi ini karena Anda akan menggunakannya saat Anda mengerjakan prosedur.

> **Note**
>
> Dengan prosedur ini di macOS Big Sur, Anda hanya dapat melakukan pembaruan kecil seperti memperbarui dari macOS Big Sur 11.7.3 ke macOS Big Sur 11.7.4. Untuk macOS Monterey atau lebih tinggi, Anda dapat melakukan pembaruan perangkat lunak utama.

Untuk mengakses disk internal

1. Dari komputer lokal Anda, di Terminal, sambungkan ke instans Apple silicon Mac Anda menggunakan SSH dengan perintah berikut. Untuk informasi selengkapnya, lihat Terhubung ke instans Anda menggunakan SSH.

```
ssh -i /path/key-pair-name.pem ec2-user@instance-public-dns-name
```

2. Instal dan mulai macOS Screen Sharing menggunakan perintah berikut.

```
[ec2-user ~]$ sudo launchctl enable system/com.apple.screensharing
sudo launchctl load -w /System/Library/LaunchDaemons/
com.apple.screensharing.plist
```

3. Tetapkan kata sandi untuk `ec2-user` dengan perintah berikut. Ingat kata sandi karena Anda akan menggunakannya nanti.

```
[ec2-user ~]$ sudo /usr/bin/dscl . -passwd /Users/ec2-user
```

4. Putuskan sambungan dari instans dengan mengetik exit dan menekan return.
5. Dari komputer lokal Anda, di Terminal, hubungkan kembali ke instans Anda dengan terowongan SSH ke port VNC menggunakan perintah berikut.

```
ssh -i /path/key-pair-name.pem -L 5900:localhost:5900 ec2-user@instance-public-
dns-name
```

> **Note**
>
> Jangan keluar dari sesi SSH ini sampai koneksi VNC berikut dan langkah-langkah GUI selesai. Ketika instans dimulai ulang, koneksi akan menutup secara otomatis.

6. Dari komputer lokal Anda, hubungkan ke `localhost:5900` menggunakan langkah-langkah berikut:
   a. Buka Finder dan pilih Go.
   b. Pilih Hubungkan ke Server.
   c. Di bidang Alamat Server, masukkan `vnc://localhost:5900`.
7. Di jendela macOS, sambungkan ke sesi jarak jauh instans Apple silicon Mac seperti kata sandi `ec2-user` yang Anda buat di Langkah 3.
8. Akses disk internal, bernama  InternalDisk, menggunakan salah satu opsi berikut.
   a. Untuk macOS Ventura atau di atasnya: Buka Pengaturan Sistem, pilih Umum di panel kiri, lalu Startup Disk di kanan bawah panel.

b. Untuk macOS Monterey atau di bawahnya: Buka Preferensi Sistem, pilih Mulai Disk, lalu buka kunci panel dengan memilih ikon kunci di kiri bawah jendela.

> **Tip pemecahan masalah**
>
> Jika Anda perlu memasang disk internal, jalankan perintah berikut di Terminal.
>
> ```
> APFSVolumeName="InternalDisk" ; SSDContainer=$(diskutil list | grep
>  "Physical Store disk0" -B 3 | grep "/dev/disk" | awk {'print $1'} ) ;
>  diskutil apfs addVolume $SSDContainer APFS $APFSVolumeName
> ```

9. Pilih disk internal, bernama  InternalDisk, dan pilih  Restart. Pilih Mulai Ulang lagi saat diminta.

> **Important**
>
> Jika disk internal diberi nama  Macintosh HD  InternalDisk, instans Anda perlu dihentikan dan dimulai ulang sehingga host khusus dapat diperbarui. Untuk informasi selengkapnya, lihat Menghentikan atau menghentikan instans Mac Amazon EC2 Anda.

Gunakan prosedur berikut untuk mendelegasikan kepemilikan kepada pengguna administratif. Ketika Anda menyambung kembali ke instans Anda dengan SSH, lakukan boot dari disk internal menggunakan user administratif khusus (`aws-managed-user`). Kata sandi awal untuk `aws-managed-user` adalah kosong, jadi Anda perlu menimpanya pada koneksi pertama Anda. Kemudian, Anda perlu mengulangi langkah-langkah untuk menginstal dan memulai MacOS Screen Sharing karena volume boot telah berubah.

Untuk mendelegasikan kepemilikan kepada administrator pada volume Amazon EBS

1. Dari komputer lokal Anda, di Terminal, sambungkan ke instans Apple silicon Mac Anda menggunakan perintah berikut.

   ```
   ssh -i /path/key-pair-name.pem aws-managed-user@instance-public-dns-name
   ```

2. Saat Anda menerima peringatan `WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!`, gunakan salah satu perintah berikut untuk mengatasi masalah ini.

a. Hapus host yang dikenal menggunakan perintah berikut. Kemudian, ulangi langkah sebelumnya.

```
rm ~/.ssh/known_hosts
```

b. Tambahkan berikut ini ke perintah SSH di langkah sebelumnya.

```
-o UserKnownHostsFile=/dev/null -o StrictHostKeyChecking=no
```

3. Tetapkan kata sandi untuk `aws-managed-user` dengan perintah berikut. Kata sandi awal `aws-managed-user` adalah kosong, jadi Anda perlu menimpanya pada koneksi pertama Anda.

a.
```
[aws-managed-user ~]$ sudo /usr/bin/dscl . -passwd /Users/aws-managed-user password
```

b. Saat Anda menerima prompt `Permission denied. Please enter user's old password:`, tekan enter.

> **Tip pemecahan masalah**
>
> Jika Anda mendapatkan kesalahan `passwd: DS error: eDSAuthFailed`, gunakan perintah berikut.
>
> ```
> [aws-managed-user ~]$ sudo passwd aws-managed-user
> ```

4. Instal dan mulai macOS Screen Sharing menggunakan perintah berikut.

```
[aws-managed-user ~]$ sudo launchctl enable system/com.apple.screensharing
sudo launchctl load -w /System/Library/LaunchDaemons/
com.apple.screensharing.plist
```

5. Putuskan sambungan dari instans dengan mengetik exit dan menekan return.

6. Dari komputer lokal Anda, di Terminal, hubungkan kembali ke instans Anda dengan terowongan SSH ke port VNC menggunakan perintah berikut.

```
ssh -i /path/key-pair-name.pem -L 5900:localhost:5900 aws-managed-user@instance-public-dns-name
```

7. Dari komputer lokal Anda, hubungkan ke `localhost:5900` menggunakan langkah-langkah berikut:

   a. Buka Finder dan pilih Go.

   b. Pilih Hubungkan ke Server.

   c. Di bidang Alamat Server, masukkan `vnc://localhost:5900`.

8. Di jendela macOS, sambungkan ke sesi jarak jauh instans Apple silicon Mac seperti kata sandi `aws-managed-user` yang Anda buat di Langkah 3.

   > **Note**
   >
   > Saat diminta untuk masuk dengan ID Apple Anda, pilih Atur Nanti.

9. Akses volume Amazon EBS menggunakan salah satu opsi berikut.

   a. Untuk macOS Ventura atau lebih baru: Buka Pengaturan Sistem, pilih Umum di panel kiri, lalu Startup Disk di kanan bawah panel.

   b. Untuk macOS Monterey atau sebelumnya: Buka Preferensi Sistem, pilih Mulai Disk, lalu buka kunci panel dengan memilih ikon kunci di kiri bawah jendela.

   > **Note**
   >
   > Sampai reboot berlangsung, ketika dimintai kata sandi administrator, gunakan kata sandi yang Anda tetapkan di atas untuk `aws-managed-user`. Kata sandi ini mungkin berbeda dari yang Anda tetapkan untuk `ec2-user` atau akun administrator default pada instans Anda. Petunjuk berikut menentukan kapan harus menggunakan kata sandi administrator instans Anda.

10. Pilih volume Amazon EBS (volume yang tidak disebutkan InternalDisk di jendela Startup Disk) dan pilih Rest art.

    > **Note**
    >
    > Jika Anda memiliki beberapa volume Amazon EBS yang dapat di-boot yang terpasang pada instans Apple silicon Mac Anda, pastikan untuk menggunakan nama unik untuk setiap volume.

11. Konfirmasikan mulai ulang, lalu pilih Otorisasi Pengguna saat diminta.
12. Pada panel Otorisasi pengguna pada volume ini, pastikan bahwa pengguna administratif (`ec2-user` secara default) telah dipilih, lalu pilih Otorisasi.
13. Masukkan `ec2-user` kata sandi yang Anda buat di Langkah 3 dari prosedur sebelumnya, lalu pilih Lanjutkan.
14. Masukkan kata sandi untuk pengguna administratif khusus (`aws-managed-user`) saat diminta.
15. Dari komputer lokal Anda, di Terminal, sambungkan kembali ke instance Anda menggunakan SSH dengan nama penggunaec2-user.

    > **Tip pemecahan masalah**
    >
    > Jika Anda mendapatkan peringatan `WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!`, jalankan perintah berikut dan hubungkan kembali ke instans Anda menggunakan SSH.
    >
    > ```
    > rm ~/.ssh/known_hosts
    > ```

16. Untuk melakukan pembaruan perangkat lunak, gunakan perintah di bawah Perbarui perangkat lunak pada instans Mac x86.

## Perbarui perangkat lunak pada instans Mac x86

Pada instans x86, Anda dapat menginstal pembaruan sistem operasi dari Apple menggunakan perintah.`softwareupdate`

Untuk menginstal pembaruan sistem operasi dari Apple pada instans Mac x86

1. Daftar paket dengan pembaruan yang tersedia menggunakan perintah berikut.

   ```
   [ec2-user ~]$ softwareupdate --list
   ```

2. Menginstal semua pembaruan atau hanya pembaruan tertentu. Untuk menginstal pembaruan tertentu, gunakan perintah berikut.

   ```
   [ec2-user ~]$ sudo softwareupdate --install label
   ```

Untuk menginstal semua pembaruan, gunakan perintah berikut.

```
[ec2-user ~]$ sudo softwareupdate --install --all --restart
```

Administrator sistem dapat menggunakannya AWS Systems Manager untuk meluncurkan pembaruan sistem operasi yang telah disetujui sebelumnya pada instans Mac x86. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS Systems Manager.

Anda dapat menggunakan Homebrew untuk menginstal pembaruan paket di AMI EC2 macOS, sehingga Anda memiliki versi terbaru dari paket ini di instans Anda. Anda juga dapat menggunakan Homebrew untuk menginstal dan menjalankan aplikasi macOS umum di Amazon EC2 macOS. Untuk informasi selengkapnya, lihat Dokumentasi Homebrew.

Untuk menginstal pembaruan menggunakan Homebrew

1. Perbarui Homebrew menggunakan perintah berikut.

   ```
   [ec2-user ~]$ brew update
   ```

2. Daftar paket dengan pembaruan yang tersedia menggunakan perintah berikut.

   ```
   [ec2-user ~]$ brew outdated
   ```

3. Menginstal semua pembaruan atau hanya pembaruan tertentu. Untuk menginstal pembaruan tertentu, gunakan perintah berikut.

   ```
   [ec2-user ~]$ brew upgrade package name
   ```

   Untuk menginstal semua pembaruan, gunakan perintah berikut.

   ```
   [ec2-user ~]$ brew upgrade
   ```

## Meningkatkan ukuran volume EBS di instans Mac Anda

Anda dapat meningkatkan ukuran volume Amazon EBS pada instans Mac Anda. Untuk informasi selengkapnya, lihat Amazon EBS Elastic Volumes di Panduan Pengguna Amazon EBS.

Setelah Anda meningkatkan ukuran volumenya, Anda harus meningkatkan ukuran kontainer APFS sebagai berikut.

Tingkatkan ruang disk yang tersedia untuk digunakan

1. Menentukan apakah mulai ulang diperlukan. Jika Anda mengubah ukuran volume EBS pada instans Mac yang sedang berjalan, Anda harus melakukan boot ulang pada instans tersebut agar volume baru tersedia. Jika modifikasi ruang disk dilakukan selama waktu peluncuran, maka boot ulang tidak akan diperlukan.

   Lihat status ukuran disk saat ini:

   ```
   [ec2-user ~]$  diskutil list external physical
   /dev/disk0 (external, physical):
      #:                       TYPE NAME                    SIZE       IDENTIFIER
      0:                 GUID_partition_scheme           *322.1 GB     disk0
      1:                 EFI EFI                          209.7 MB     disk0s1
      2:                 Apple_APFS Container disk2       321.9 GB     disk0s2
   ```

2. Salin dan tempel perintah berikut ini.

   ```
   [ec2-user ~]$ PDISK=$(diskutil list physical external | head -n1 | cut -d" " -f1)
   APFSCONT=$(diskutil list physical external | grep "Apple_APFS" | tr -s " " | cut -
   d" " -f8)
   yes | sudo diskutil repairDisk $PDISK
   ```

3. Salin dan tempel perintah berikut ini.

   ```
   [ec2-user ~]$ sudo diskutil apfs resizeContainer $APFSCONT 0
   ```

## Menghentikan atau menghentikan instans Mac Amazon EC2 Anda

Bila Anda menghentikan instans Mac, maka instans tersebut tetap berada di dalam status `stopping` selama sekitar 15 menit sebelum memasuki status.`stopped`

Saat Anda menghentikan atau mengakhiri instans Mac, Amazon EC2 melakukan alur kerja scrubbing pada Host Khusus dasar untuk menghapus SSD internal, untuk membersihkan variabel NVRAM persisten, dan memperbarui ke firmware perangkat terbaru. Ini memastikan bahwa instans Mac memberikan keamanan dan privasi data yang sama dengan instans EC2 Nitro lainnya. Ini juga memungkinkan Anda untuk menjalankan AMI macOS terbaru. Selama alur kerja scrubbing,

Host Khusus sementara memasuki status tertunda. Pada instans x86 Mac, alur kerja scrubbing mungkin membutuhkan waktu hingga 50 menit untuk menyelesaikannya. Jika Amazon EC2 perlu memperbarui firmware perangkat, alur kerja mungkin memakan waktu hingga 3 jam untuk menyelesaikannya. Pada instans Mac silikon Apple, alur kerja penggosokan mungkin memakan waktu hingga 4,5 jam untuk diselesaikan.

Anda tidak dapat memulai instans Mac yang dihentikan atau meluncurkan instans Mac baru sampai setelah alur kerja scrubbing selesai, yang mana pada saat itu Host Khusus memasuki status.`available`

Pengukuran dan penagihan dijeda saat Host Khusus memasuki status.`pending` Anda tidak dikenai biaya untuk alur kerja scrubbing.

Lepaskan Host Khusus untuk instans Mac Anda

Saat Anda selesai dengan instans Mac, Anda dapat melepaskan Host Khusus. Sebelum Anda dapat melepaskan Host Khusus, Anda harus menghentikan atau mengakhiri instans Mac. Anda tidak dapat melepaskan host hingga periode alokasi melebihi minimal 24 jam.

Untuk melepas Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans dan pilih Status instans, lalu pilih Hentikan instans atau Akhiri instans.
4. Di panel navigasi, pilih Host Khusus.
5. Pilih Host Khusus dan pilih Tindakan, Lepas host.
6. Ketika diminta konfirmasi, pilih Lepaskan.

## Konfigurasikan Perlindungan Integritas Sistem untuk instans Mac Amazon EC2

Anda dapat mengonfigurasi pengaturan System Integrity Protection (SIP) untuk instans Mac x86 dan instans Mac silikon Apple. SIP adalah fitur keamanan macOS penting yang membantu mencegah eksekusi kode yang tidak sah dan modifikasi tingkat sistem. Untuk informasi selengkapnya, lihat Tentang Perlindungan Integritas Sistem.

Anda dapat mengaktifkan atau menonaktifkan SIP sepenuhnya, atau Anda dapat secara selektif mengaktifkan atau menonaktifkan pengaturan SIP tertentu. Disarankan agar Anda menonaktifkan SIP hanya sementara untuk melakukan tugas-tugas yang diperlukan, dan kemudian mengaktifkannya

kembali sesegera mungkin. Membiarkan SIP dinonaktifkan dapat membuat instance Anda rentan terhadap kode berbahaya.

Konfigurasi SIP didukung di semua Wil AWS ayah di mana instans Mac Amazon EC2 didukung.

Topik


- Pertimbangan-pertimbangan
- Konfigurasi SIP default
- Periksa konfigurasi SIP Anda
- Prasyarat untuk instans Mac silikon Apple
- Konfigurasikan pengaturan SIP
- Periksa status tugas konfigurasi SIP


## Pertimbangan-pertimbangan

- Jenis instans Amazon EC2 Mac dan versi macOS berikut didukung:
  - Mac1 | Mac2 | Mac2-m1ultra - macOS Ventura (versi 13.0 atau lebih baru)
  - Mac2-m2 | Mac2-m2pro - macOS Ventura (versi 13.2 atau lebih baru)
  - Mac-m4 | Mac-m4pro - macOS Sequoia (versi 15.6 atau lebih baru)

  > **Note**
  >
  > Versi beta dan pratinjau macOS tidak didukung.

- Anda dapat menentukan konfigurasi SIP kustom untuk secara selektif mengaktifkan atau menonaktifkan pengaturan SIP individual. Jika Anda menerapkan konfigurasi khusus, sambungkan ke instance dan verifikasi pengaturan untuk memastikan bahwa persyaratan Anda diimplementasikan dengan benar dan berfungsi sebagaimana dimaksud.

  Konfigurasi SIP mungkin berubah dengan pembaruan macOS. Kami menyarankan Anda meninjau pengaturan SIP khusus setelah peningkatan versi macOS untuk memastikan kompatibilitas berkelanjutan dan fungsionalitas konfigurasi keamanan yang tepat.
- Untuk instans Mac x86, pengaturan SIP diterapkan pada tingkat instance. Setiap volume root yang dilampirkan ke instance akan secara otomatis mewarisi pengaturan SIP yang dikonfigurasi.

Untuk instance Mac silikon Apple, pengaturan SIP diterapkan pada tingkat volume. Volume root yang dilampirkan ke instance tidak mewarisi pengaturan SIP. Jika Anda melampirkan volume root lain, Anda harus mengkonfigurasi ulang pengaturan SIP ke status yang diperlukan.

- Diperlukan waktu hingga 90 menit untuk menyelesaikan tugas konfigurasi SIP. Instans tetap tidak dapat dijangkau saat tugas konfigurasi SIP sedang berlangsung.
- Konfigurasi SIP tidak ditransfer ke snapshot atau AMI yang kemudian Anda buat dari instance.
- Instans Mac silikon Apple hanya harus memiliki satu volume yang dapat di-boot, dan setiap volume terpasang hanya dapat memiliki satu pengguna admin tambahan.

### Konfigurasi SIP default

Tabel berikut mencantumkan konfigurasi SIP default untuk instans Mac x86 dan instans Mac silikon Apple.

| | Instans Mac silikon Apple | Instans Mac x86 |
|---|---|---|
| Apple Internal | Diaktifkan | Dinonaktifkan |
| Perlindungan Sistem File | Diaktifkan | Dinonaktifkan |
| Sistem Dasar | Diaktifkan | Diaktifkan |
| Pembatasan Debugging | Diaktifkan | Diaktifkan |
| Pembatasan Dtrace | Diaktifkan | Diaktifkan |
| Penandatanganan Kext | Diaktifkan | Diaktifkan |
| Perlindungan Nvram | Diaktifkan | Diaktifkan |

### Periksa konfigurasi SIP Anda

Kami menyarankan Anda memeriksa konfigurasi SIP Anda sebelum dan sesudah membuat perubahan untuk memastikan bahwa konfigurasi tersebut dikonfigurasi seperti yang diharapkan.

Untuk memeriksa konfigurasi SIP untuk instans Mac Amazon EC2

Hubungkan ke instance menggunakan SSH, dan kemudian jalankan perintah berikut di baris perintah.

```
$ csrutil status
```

Berikut ini adalah output contoh.

```
System Integrity Protection status: enabled.

Configuration:
    Apple Internal: enabled
    Kext Signing: disabled
    Filesystem Protections: enabled
    Debugging Restrictions: enabled
    DTrace Restrictions: enabled
    NVRAM Protections: enabled
    BaseSystem Verification: disabled
```

Prasyarat untuk instans Mac silikon Apple

Sebelum Anda dapat mengonfigurasi pengaturan SIP untuk instans Mac silikon Apple, Anda harus menetapkan kata sandi dan mengaktifkan token aman untuk pengguna administratif volume root Amazon EBS (`ec2-user`).

> **Note**
>
> Kata sandi dan token aman diatur saat pertama kali Anda terhubung ke instance Mac silikon Apple menggunakan GUI. Jika sebelumnya Anda terhubung ke instance menggunakan GUI, atau jika Anda menggunakan instance x86 Mac, Anda tidak perlu melakukan langkah-langkah ini.

> **Note**
>
> Semua nama pengguna dan kata sandi macOS yang digunakan untuk otentikasi macOS harus antara 4 dan 16 karakter untuk digunakan dengan panggilan API pengaturan SIP.

Untuk mengatur kata sandi dan mengaktifkan token aman untuk pengguna administratif volume root EBS

1. Hubungkan ke instance menggunakan SSH.
2. Tetapkan kata sandi untuk `ec2-user` pengguna.

```
$ sudo /usr/bin/dscl . -passwd /Users/ec2-user
```

3. Aktifkan token aman untuk `ec2-user` pengguna. Untuk `-oldPassword`, tentukan kata sandi yang sama dari langkah sebelumnya. Untuk `-newPassword`, tentukan kata sandi yang berbeda. Perintah berikut mengasumsikan bahwa Anda memiliki kata sandi lama dan baru yang disimpan dalam `.txt` file.

```
$ sysadminctl -oldPassword `cat old_password.txt` -newPassword `cat
 new_password.txt`
```

4. Verifikasi bahwa token aman diaktifkan.

```
$ sysadminctl -secureTokenStatus ec2-user
```

Konfigurasikan pengaturan SIP

Saat mengonfigurasi pengaturan SIP untuk instance Anda, Anda dapat mengaktifkan atau menonaktifkan semua pengaturan SIP, atau Anda dapat menentukan konfigurasi khusus yang secara selektif mengaktifkan atau menonaktifkan pengaturan SIP tertentu.

> **Note**
>
> Jika Anda menerapkan konfigurasi khusus, sambun gkan ke instance dan verifikasi pengaturan untuk memastikan bahwa persyaratan Anda diimplementasikan dengan benar dan berfungsi sebagaimana dimaksud.
> Konfigurasi SIP mungkin berubah dengan pembaruan macOS. Kami menyarankan Anda meninjau pengaturan SIP khusus setelah peningkatan versi macOS untuk memastikan kompatibilitas berkelanjutan dan fungsionalitas konfigurasi keamanan yang tepat.

Untuk mengkonfigurasi pengaturan SIP untuk instance Anda, Anda harus membuat tugas konfigurasi SIP. Tugas konfigurasi SIP menentukan pengaturan SIP untuk instance Anda.

Saat membuat konfigurasi SIP untuk instance Mac silikon Apple, Anda harus menentukan kredenSIAL berikut:

- Pengguna administratif disk internal
  - Nama pengguna — Hanya pengguna administratif default (`aws-managed-user`) yang didukung dan digunakan secara default. Anda tidak dapat menentukan pengguna administratif yang berbeda.
  - Kata Sandi — Jika Anda tidak mengubah kata sandi default untuk`aws-managed-user`, tentukan kata sandi default, yang kosong. Jika tidak, tentukan kata sandi Anda.
- Pengguna administratif volume root Amazon EBS
  - Nama Pengguna — Jika Anda tidak mengubah pengguna administratif default, tentukan`ec2-user`. Jika tidak, tentukan nama pengguna untuk pengguna administratif Anda.
  - Kata Sandi — Anda harus selalu menentukan kata sandi.

Gunakan metode berikut untuk membuat tugas konfigurasi SIP.

Console

Untuk membuat tugas konfigurasi SIP menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih instans Amazon EC2 Mac.
3. Di tab Keamanan, pilih Ubah Mac, Ubah Perlindungan Integritas Sistem.
4. Untuk mengaktifkan semua pengaturan SIP, pilih Aktifkan SIP. Untuk menonaktifkan semua pengaturan SIP, hapus Aktifkan SIP.
5. Untuk menentukan konfigurasi kustom yang secara selektif mengaktifkan atau menonaktifkan pengaturan SIP tertentu, pilih T entukan konfigurasi SIP kustom, lalu pilih pengaturan SIP untuk diaktifkan, atau hapus pengaturan SIP untuk dinonaktifkan.
6. Tentukan kredenSIAL untuk pengguna volume root dan pemilik disk internal.
7. Pilih Buat tugas modifikasi SIP.

AWS CLI

Untuk membuat tugas konfigurasi SIP menggunakan AWS CLI

Gunakan perintah create-mac- system-integrity-protection-modifiation-task.

Mengaktifkan atau menonaktifkan semua pengaturan SIP

Untuk sepenuhnya mengaktifkan atau menonaktifkan semua pengaturan SIP, gunakan hanya `--mac-system-integrity-protection-status` parameter.

Perintah contoh berikut mengaktifkan semua pengaturan SIP.

```
aws ec2 create-mac-system-integrity-protection-modification-task \
--instance-id i-0abcdef9876543210 \
--mac-system-integrity-protection-status enabled \
--mac-credentials file://mac-credentials.json
```

Tentukan konfigurasi SIP kustom

Untuk menentukan konfigurasi SIP kustom yang secara selektif mengaktifkan atau menonaktifkan pengaturan SIP tertentu, tentukan `--mac-system-integrity-protection-configuration` parameter `--mac-system-integrity-protection-status` dan. Dalam hal ini, gunakan `mac-system-integrity-protection-status` untuk menentukan status SIP keseluruhan, dan gunakan `mac-system-integrity-protection-configuration` untuk secara selektif mengaktifkan atau menonaktifkan pengaturan SIP individual.

Perintah contoh berikut membuat tugas konfigurasi SIP untuk mengaktifkan semua pengaturan SIP, kecuali `NvramProtections` dan`FilesystemProtections`.

```
aws ec2 create-mac-system-integrity-protection-modification-task \
--instance-id i-0abcdef9876543210 \
--mac-system-integrity-protection-status enabled \
--mac-system-integrity-protection-configuration
 "NvramProtections=disabled, FilesystemProtections=disabled" \
--mac-credentials file://mac-credentials.json
```

Perintah contoh berikut membuat tugas konfigurasi SIP untuk menonaktifkan semua pengaturan SIP, kecuali`DtraceRestrictions`.

```
aws ec2 create-mac-system-integrity-protection-modification-task \
--instance-id i-0abcdef9876543210 \
--mac-system-integrity-protection-status disabled \
--mac-system-integrity-protection-configuration "DtraceRestrictions=enabled" \
--mac-credentials file://mac-credentials.json
```

Isi file **`mac-credentials.json`**

Berikut ini adalah isi `mac-credentials.json` file yang direferensikan dalam contoh sebelumnya.

```
{
  "internalDiskPassword":"internal-disk-admin_password",
  "rootVolumeUsername":"root-volume-admin_username",
  "rootVolumePassword":"root-volume-admin_password"
}
```

Periksa status tugas konfigurasi SIP

Gunakan salah satu metode berikut untuk memeriksa status tugas konfigurasi SIP.

Console

Untuk melihat tugas konfigurasi SIP menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih instans Amazon EC2 Mac.
3. Di tab Keamanan, gulir ke bawah ke bagian tugas modifikasi Mac.

AWS CLI

Untuk memeriksa status tugas konfigurasi SIP menggunakan AWS CLI

Gunakan perintah describe-mac-modifiation-tasks.

## Temukan versi macOS yang didukung untuk Amazon EC2 Mac Dedicated Host

Anda dapat melihat versi macOS terbaru yang didukung oleh Amazon EC2 Mac Dedicated Host Anda. Dengan fungsi ini, Anda dapat memvalidasi apakah Host Khusus dapat mendukung peluncuran instance dengan versi macOS pilihan Anda.

Setiap versi macOS memerlukan versi firmware minimum pada Apple Mac yang mendasarinya agar berhasil boot. Versi firmware Apple Mac dapat menjadi usang jika Host Khusus Mac yang dialokasikan tetap idle untuk jangka waktu yang lama atau jika memiliki instance yang berjalan lama di atasnya.

Untuk memastikan dukungan untuk versi macOS terbaru, Anda dapat menghentikan atau menghentikan instans di Host Khusus Mac yang dialokasikan. Ini memicu alur kerja penggosokan host dan memperbarui firmware pada Apple Mac yang mendasarinya untuk mendukung versi macOS terbaru. Host Khusus dengan instans yang berjalan lama akan diperbarui secara otomatis saat Anda menghentikan atau menghentikan instans yang sedang berjalan.

Untuk informasi selengkapnya tentang alur kerja menggosok, lihatMenghentikan atau menghentikan instans Mac Amazon EC2 Anda.

Untuk informasi selengkapnya tentang meluncurkan instans Mac, lihatLuncurkan instans Mac menggunakan konsol Konsol Manajemen AWS atau AWS CLI.

Anda dapat melihat informasi tentang versi macOS terbaru yang didukung di Host Khusus yang dialokasikan menggunakan konsol Amazon EC2 atau. AWS CLI

Console

Untuk melihat informasi firmware Host Khusus menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Di halaman detail Host Khusus, di bawah Versi macOS terbaru yang didukung, Anda dapat melihat versi macOS terbaru yang dapat didukung oleh host.

AWS CLI

Untuk melihat informasi firmware Host Khusus menggunakan AWS CLI

Gunakan `describe-mac-hosts` perintah, ganti `region` dengan yang sesuai Wilayah AWS.

```
$ aws ec2 describe-mac-hosts --region us-east-1
  {
      "MacHosts": [
          {
              "HostId": "h-07879acf49EXAMPLE",
              "MacOSLatestSupportedVersions": [
                  "14.3",
                  "13.6.4",
                  "12.7.3"
              ]
```

```
            }
        ]
    }
```

## Berlangganan notifikasi AMI macOS

Agar mendapatkan notifikasi saat AMI baru dirilis atau saat bridgeOS diperbarui, berlangganan notifikasi menggunakan Amazon SNS.

Untuk informasi selengkapnya tentang AMI macOS EC2, lihatCatatan rilis AMI macOS Amazon EC2.

Untuk berlangganan notifikasi AMI macOS

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di bilah navigasi, ubah Wilayah menjadi AS Timur (Virginia Utara), jika perlu. Anda harus memilih Wilayah ini karena notifikasi SNS tempat Anda berlangganan ada di Wilayah ini.
3. Di panel navigasi, pilih Langganan.
4. Pilih Buat langganan.
5. Untuk kotak dialog Buat langganan, lakukan hal berikut:
   a. Untuk Topik ARN, salin dan tempel salah satu Amazon Resource Name (ARN) berikut:
      - **`arn:aws:sns:us-east-1:898855652048:amazon-ec2-macos-ami-updates`**
      - **`arn:aws:sns:us-east-1:898855652048:amazon-ec2-bridgeos-updates`**
   b. Untuk  Protokol, pilih salah satu dari berikut ini:
      - Email:

        Untuk Titik Akhir, ketik alamat email yang bisa Anda gunakan untuk menerima notifikasi. Setelah Anda berlangganan, Anda akan menerima pesan konfirmasi dengan baris subjek `AWS Notification - Subscription Confirmation`. Buka email dan pilih Konfirmasi berlangganan untuk menyelesaikan langganan Anda
      - SMS:

        Untuk Titik Akhir, ketik nomor telepon yang bisa Anda gunakan untuk menerima notifikasi.
      - AWS Lambda, Amazon SQS, Amazon Data Firehose  (Pember  itahuan datang dalam format  JSON):

Untuk Titik akhir, masukkan ARN untuk fungsi Lambda, antrean SQS, atau aliran Firehose yang dapat Anda gunakan untuk menerima notifikasi.

c. Pilih Buat langganan.

Kapan pun AMI macOS dirilis, kami akan mengirimkan notifikasi kepada pelanggan topik.`amazon-ec2-macos-ami-updates` Kapan pun bridgeOS diperbarui, kami akan mengirimkan notifikasi kepada pelanggan topik.`amazon-ec2-bridgeos-updates` Jika Anda tidak ingin lagi menerima notifikasi ini, gunakan prosedur berikut untuk berhenti berlangganan.

Untuk membatalkan langganan notifikasi AMI macOS

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di bilah navigasi, ubah Wilayah menjadi AS Timur (Virginia Utara), jika perlu. Anda harus memilih Wilayah ini karena notifikasi SNS dibuat di Wilayah ini.
3. Di panel navigasi, pilih Langganan.
4. Pilih langganan, lalu pilih Tindakan, Hapus langganan. Saat diminta konfirmasi, pilih Hapus.

## Ambil ID AMI macOS menggunakan AWS Systems Manager API Penyimpanan Parameter

Anda harus menentukan AMI saat meluncurkan instans. AMI khusus untuk sistem operasi Wilayah AWS, dan arsitektur prosesor. Anda dapat melihat semua AMI macOS di Wilayah AWS dan mengambil AMI macOS terbaru dengan menanyakan API Penyimpanan AWS Systems Manager Parameter. Dengan menggunakan parameter publik ini, Anda tidak perlu mencari ID AMI macOS secara manual. Parameter publik tersedia untuk kedua x86 dan ARM64 macOS AMI, dan dapat diintegrasikan dengan AWS CloudFormation templat yang ada.

Izin yang diperlukan

Untuk melakukan tindakan ini, prinsi pal IAM harus memiliki izin untuk memanggil tindakan `ssm:GetParameter` API.

Untuk melihat daftar semua AMI macOS di saat ini Wilayah AWS menggunakan AWS CLI

Gunakan perintah get-parameters-by-path berikut untuk melihat daftar semua AMI macOS di Wilayah saat ini.

```
aws ssm get-parameters-by-path --path /aws/service/ec2-macos --recursive --query
 "Parameters[].Name"
```

Untuk mengambil ID AMI dari AMI macOS utama terbaru menggunakan AWS CLI

Gunakan perintah get-parameter berikut dengan sub-parameter. `image_id` Dalam contoh berikut, ganti `sonoma` dengan versi utama yang didukung macOS, `x86_64_mac` dengan prosesor, dan `region-code` dengan Wilayah AWS dukungan yang Anda inginkan ID AMI macOS terbaru.

```
aws ssm get-parameter --name /aws/service/ec2-macos/sonoma/x86_64_mac/latest/image_id
 --region region-code
```

Untuk informasi selengkapnya, lihat Memang gil parameter publik AMI untuk macOS di Panduan AWS Systems Manager Pengguna.

## Catatan rilis AMI macOS Amazon EC2

Informasi berikut memberikan rincian tentang paket yang disertakan secara default di AMI EC2 macOS dan merangkum perubahan untuk setiap rilis EC2 macOS AMI.

Untuk informasi tentang cara berlangganan notifikasi AMI macOS, lihatBerlangganan notifikasi AMI macOS.

Instans Mac dapat menjalankan salah satu sistem operasi berikut:

- macOS Mojave (versi 10.14) (hanya instans x86 Mac)
- macOS Catalina (versi 10.15) (hanya instans x86 Mac)
- macOS Big Sur (versi 11) (instans x86 dan M1 Mac)
- macOS Monterey (versi 12) (instans x86 dan M1 Mac)
- macOS Ventura (versi 13) (semua instans Mac, instans M2 dan M2 Pro Mac mendukung macOS Ventura versi 13.2 atau setelahnya)
- macOS Sonoma (versi 14) (semua instans Mac)
- macOS Sequoia (versi 15) (semua contoh Mac)

> **Note**
>
> Instans Mac M4 dan M4 Pro mendukung macOS Sequoia versi 15.6 atau yang lebih baru.

### Menyetujui Kebijakan Privasi Jaringan Lokal untuk macOS Sequoia

macOS Sequoia (versi 15) memiliki fitur Privasi Jaringan Lokal baru yang berdampak pada pengguna IP-based layanan lokal, termasuk Amazon EC2 Instance Metadata Service (IMDS).

> **Important**
>
> Untuk memastikan bahwa Anda memiliki akses tanpa gangguan ke IP-based layanan lokal, gunakan langkah-langkah berikut untuk menyetujui kebijakan Privasi Jaringan Lokal.

Untuk menyetujui Kebijakan Privasi Jaringan Lokal

1. Hubungkan ke antarmuka pengguna grafis (GUI) instans Anda.
2. Ikuti petunjuk di layar untuk menyetujui kebijakan Privasi Jaringan Lokal.
3. Setelah Anda menyetujui kebijakan, buat AMI instans EC2 Mac Anda. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.

Instans EC2 Mac yang diluncurkan dari AMI yang baru dibuat akan mempertahankan izin Privasi Jaringan Lokal.

### Paket default termasuk dalam Amazon EC2 macOS AMI

Tabel berikut menjelaskan paket yang disertakan secara default di AMI macOS EC2.

| Paket | Catatan rilis |
| --- | --- |
| EC2Inisiasi macOS | https://github.com/aws/ec2-macos-init/tags |
| EC2Utilitas macOS | https://github.com/aws/ec2-macos-utils/tags |
| AmazonAgen SSM | https://github.com/aws/amazon-ssm-agent/releases |
| AWS Command Line Interface (AWS CLI) versi 2 | https://raw.githubusercontent.com/aws/aws-cli/v2/CHANGELOG.rst |
| Alat Baris Perintah untuk Xcode | https://developer.apple.com/documentation/xcode-release-notes |

| Paket | Catatan rilis |
| --- | --- |
| Homebrew | https://github.com/Homebrew/brew/releases |
| EC2 Instance Connect | https://github.com/aws/aws-ec2-instance-connect-config/releases |
| Safari | https://developer.apple.com/documentation/safari-release-notes |

Pembaruan Amazon EC2 macOS AMI

Tabel berikut menjelaskan perubahan yang disertakan dalam rilis EC2 macOS AMI. Perhatikan bahwa beberapa perubahan berlaku untuk semua AMI macOS EC2, sedangkan yang lain hanya berlaku untuk subset dari AMI ini.

Pembaruan EC2 macOS AMI

| Rilis | Perubahan |
| --- | --- |
| 2026.09.02 | Semua AMI<br>• Diperbarui `awscli` ke 2.36.36<br>• Memperbarui Homebrew ke 6.0.20<br><br>Rilis macOS Sonoma 14.8.9 (silikon Apple)<br>• Konten keamanan macOS Sonoma 14.8.9<br>• Diperbarui Safari ke 26.6.1<br>  • Konten keamanan Safari 26.6.1<br><br>Rilis macOS Tahoe 26.6.2<br>• Konten keamanan macOS Tahoe 26.6.2 |
| 2026.08.26 | Semua AMI<br>• Diperbarui `awscli` ke 2.36.27 |

| Rilis | Perubahan |
| --- | --- |
| | • Memperbarui Homebrew ke 6.0.18<br><br>Rilis macOS Sonoma 14.8.9 (hanya Intel, **`mac1.metal`**)<br><br>• Konten keamanan macOS Sonoma 14.8.9<br>• Diperbarui Safari ke 26.6.1<br>  • Konten keamanan Safari 26.6.1<br><br>Rilis macOS Sequoia 15.7.9<br><br>• Konten keamanan macOS Sequoia 15.7.9<br>• Diperbarui Safari ke 26.6.1<br>  • Konten keamanan Safari 26.6.1<br><br>Rilis macOS Tahoe 26.6.1<br><br>• Konten keamanan macOS Tahoe 26.6.1 |

| Rilis | Perubahan |
| --- | --- |
| 2026.08.13 | Semua AMI<br>• Diperbarui `awscli` ke 2.36.12<br>• Memperbarui Homebrew ke 6.0.13<br><br>Rilis macOS Sonoma 14.8.8 (hanya Intel, **`mac1.metal`**)<br>• Konten keamanan macOS Sonoma 14.8.8<br>• Diperbarui Safari ke 26.6<br>  • Konten keamanan Safari 26.6<br><br>Rilis macOS Sequoia 15.7.8<br>• Konten keamanan macOS Sequoia 15.7.8<br>• Diperbarui Safari ke 26.6<br>  • Konten keamanan Safari 26.6<br><br>Rilis macOS Tahoe 26.6<br>• Konten keamanan macOS Tahoe 26.6<br>• Alat Baris Perintah yang Diperbarui ke 26.6 |
| 2026.07.29 | Rilis macOS Tahoe 26.5.2<br>• Konten keamanan macOS Tahoe 26.5.2<br>• Diperbarui `awscli` ke 2.36.1<br>• Memperbarui Homebrew ke 6.0.11 |

| Rilis | Perubahan |
| --- | --- |
| 2026.05.19 | Semua AMI<br>• Diperbarui `awscli` ke 2.34.46<br>• Memperbarui Homebrew ke 5.1.11<br><br>Rilis macOS Sonoma 14.8.7<br>• Konten keamanan macOS Sonoma 14.8.7<br>• Diperbarui Safari ke 26.5<br>  • Konten keamanan Safari 26.5<br><br>Rilis macOS Sequoia 15.7.7<br>• Konten keamanan macOS Sequoia 15.7.7<br>• Diperbarui Safari ke 26.5<br>  • Konten keamanan Safari 26.5<br><br>Rilis macOS Tahoe 26.5<br>• Konten keamanan macOS Tahoe 26.5<br>• Alat Baris Perintah yang Diperbarui ke 26.5 |
| 2026.04.20 | Rilis macOS Tahoe 26.4.1<br>• Konten keamanan macOS Tahoe 26.4.1<br>• Alat Baris Perintah yang Diperbarui ke 26.4.1<br>• Diperbarui `awscli` ke 2.34.32<br>• Memperbarui Homebrew ke 5.1.7 |

| Rilis | Perubahan |
| --- | --- |
| 2026.04.16 | Semua AMI<br>• Diperbarui `awscli` ke 2.34.27<br>• Diperbarui `ec2-macos-init` ke 1.5.15<br>• Memperbarui Homebrew ke 5.1.5<br><br>Rilis macOS Sonoma 14.8.5<br>• Konten keamanan macOS Sonoma 14.8.5<br>• Diperbarui Safari ke 26.4<br>  • Konten keamanan Safari 26.4<br><br>Rilis macOS Sequoia 15.7.5<br>• Konten keamanan macOS Sequoia 15.7.5<br>• Diperbarui Safari ke 26.4<br>  • Konten keamanan Safari 26.4<br><br>Rilis macOS Tahoe 26.4<br>• Konten keamanan macOS Tahoe 26.4<br>• Alat Baris Perintah yang Diperbarui ke 26.4 |
| 2026.03.17 | Rilis macOS Tahoe 26.3.1<br>• Apa yang baru dalam pembaruan untuk macOS Tahoe 26<br>• Alat Baris Perintah yang Diperbarui ke 26.3<br>• Diperbarui `awscli` ke 2.34.10<br>• Memperbarui Homebrew ke 5.1.0 |

| Rilis | Perubahan |
| --- | --- |
| 2026.03.03 | Semua AMI<br>• Diperbarui `awscli` ke 2.33.31<br>• Memperbarui Homebrew ke 5.0.15<br><br>Rilis macOS Sonoma 14.8.4<br>• Konten keamanan macOS Sonoma 14.8.4<br>• Diperbarui Safari ke 26.3<br>  • Konten keamanan Safari 26.3<br><br>Rilis macOS Sequoia 15.7.4<br>• Konten keamanan macOS Sequoia 15.7.4<br>• Diperbarui Safari ke 26.3<br>  • Konten keamanan Safari 26.3<br><br>Rilis macOS Tahoe 26.3<br>• Konten keamanan macOS Tahoe 26.3 |

| Rilis | Perubahan |
| --- | --- |
| 2025.12.26 | Semua AMI<br>• Diperbarui `awscli` ke 2.32.19<br>• Diperbarui `amazon-ssm-agent` ke 3.3.3270.0<br>• Memperbarui Homebrew ke 5.0.6<br><br>Rilis macOS Sonoma 14.8.3<br>• Konten keamanan macOS Sonoma 14.8.3<br>• Diperbarui Safari ke 26.2<br>  • Konten keamanan Safari 26.2<br><br>Rilis macOS Sequoia 15.7.3<br>• Konten keamanan macOS Sequoia 15.7.3<br>• Diperbarui Safari ke 26.2<br>  • Konten keamanan Safari 26.2<br>• Alat Baris Perintah yang Diperbarui ke 26.2<br><br>Rilis macOS Tahoe 26.2<br>• Konten keamanan macOS Tahoe 26.2<br>• Alat Baris Perintah yang Diperbarui ke 26.2 |

| Rilis | Perubahan |
| --- | --- |
| 2025.12.17 | Semua AMI<br><br>• Diperbarui `awscli` ke 2.32.16<br>• Memperbarui Homebrew ke 5.0.5<br><br>Rilis macOS Sonoma 14.8.2<br><br>• Konten keamanan macOS Sonoma 14.8.2<br>• Diperbarui Safari ke 26.1<br>  • Konten keamanan Safari 26.1<br><br>Rilis macOS Sequoia 15.7.2<br><br>• Konten keamanan macOS Sequoia 15.7.2<br>• Diperbarui Safari ke 26.1<br>  • Konten keamanan Safari 26.1<br>• Alat Baris Perintah yang Diperbarui ke 26.1<br><br>Rilis macOS Tahoe 26.1<br><br>• Konten keamanan macOS Tahoe 26.1<br>• Alat Baris Perintah yang Diperbarui ke 26.1 |

| Rilis | Perubahan |
| --- | --- |
| 2025.11.18 | Semua AMI<br><br>• Diperbarui `awscli` ke 2.31.35<br>• Diperbarui `ec2-macos-init` ke 1.5.13<br>• Diperbarui `ec2-macos-utils` ke 1.0.7<br>• Memperbarui Homebrew ke 5.0.1<br><br>Rilis macOS Sonoma 14.8.1<br><br>• Konten keamanan macOS Sonoma 14.8.1<br>• Diperbarui Safari ke 26.0.1<br>  • Konten keamanan Safari 26.0.1<br><br>Rilis macOS Sequoia 15.7.1<br><br>• Konten keamanan macOS Sequoia 15.7.1<br>• Diperbarui Safari ke 26.0.1<br>  • Konten keamanan Safari 18.6<br>• Alat Baris Perintah yang Diperbarui ke 26.0<br><br>Rilis macOS Tahoe 26.0.1<br><br>• Konten keamanan macOS Tahoe 26.0.1<br>• Alat Baris Perintah yang Diperbarui ke 26.0 |

| Rilis | Perubahan |
| --- | --- |
| 2025.09.04 | Semua AMI<br>• Diperbarui `awscli` ke 2.28.19<br>• Diperbarui `ec2-instance-connect` ke 2.0.0-5<br>• Memperbarui Homebrew ke 4.6.7<br><br>Rilis macOS Ventura 13.7.8<br>• Konten keamanan macOS Ventura 13.7.8<br><br>Rilis macOS Sonoma 14.7.8<br>• Konten keamanan macOS Sonoma 14.7.8<br><br>Rilis macOS Sequoia 15.6.1<br>• Konten keamanan macOS Sequoia 15.6.1 |

| Rilis | Perubahan |
| --- | --- |
| 2025.08.05 | Semua AMI<br>• Diperbarui `awscli` ke 2.28.0<br>• Memperbarui Homebrew ke 4.5.13<br><br>Rilis macOS Ventura 13.7.7<br>• Konten keamanan macOS Ventura 13.7.7<br>• Diperbarui Safari ke 18.6<br>  • Konten keamanan Safari 18.6<br><br>Rilis macOS Sonoma 14.7.7<br>• Konten keamanan macOS Sonoma 14.7.7<br>• Diperbarui Safari ke 18.6<br>  • Konten keamanan Safari 18.6<br><br>Rilis macOS Sequoia 15.6<br>• Konten keamanan macOS Sequoia 15.6<br>• Diperbarui AWS ENA Ethernet ke 2.0.0 |

| Rilis | Perubahan |
| --- | --- |
| 2025.06.27 | Semua AMI<br><br>• Diperbarui `awscli` ke 2.27.40<br>• Memperbarui Homebrew ke 4.5.7<br>• Gambar x86_64 yang dimigrasikan ke AWS dext Ethernet ENA<br><br>Rilis macOS Ventura 13.7.6<br><br>• Konten keamanan macOS Ventura 13.7.6<br>• Diperbarui Safari ke 18.5<br>  • Konten keamanan Safari 18.5<br><br>Rilis macOS Sonoma 14.7.6<br><br>• Konten keamanan macOS Sonoma 14.7.6<br>• Diperbarui Safari ke 18.5<br>  • Konten keamanan Safari 18.5<br><br>Rilis macOS Sequoia 15.5<br><br>• Konten keamanan macOS Sequoia 15.5<br>• Alat Baris Perintah yang Diperbarui ke 16.4<br>• Diperbarui AWS ENA Ethernet ke 1.0.9 |

| Rilis | Perubahan |
| --- | --- |
| 2025.05.21 | Semua AMI<br>• Diperbarui `awscli` ke 2.27.7<br>• Diperbarui `ec2-macos-init` ke 1.5.11<br>• Diperbarui `ec2-macos-utils` ke 1.0.5<br>• Memperbarui Homebrew ke 4.5.1<br><br>Rilis macOS Ventura 13.7.5<br>• Konten keamanan macOS Ventura 13.7.5<br>• Diperbarui Safari ke 18.4<br>  • Konten keamanan Safari 18.4<br><br>Rilis macOS Sonoma 14.7.5<br>• Konten keamanan macOS Sonoma 14.7.5<br>• Diperbarui Safari ke 18.4<br>  • Konten keamanan Safari 18.4<br><br>Rilis macOS Sequoia 15.4.1<br>• Konten keamanan macOS Sequoia 15.4.1<br>• Alat Baris Perintah yang Diperbarui ke 16.3 |

| Rilis | Perubahan |
| --- | --- |
| 2025.05.05 | Semua AMI<br><br>• Diperbarui `awscli` ke 2.27.1<br>• Diperbarui `ec2-macos-init` ke 1.5.11<br>• Diperbarui `ec2-macos-system-monitor` ke 1.3.1<br>• Diperbarui `ec2-macos-utils` ke 1.0.5<br>• Memperbarui Homebrew ke 4.4.32<br><br>Rilis macOS Ventura 13.7.5<br><br>• Konten keamanan macOS Ventura 13.7.5<br>• Diperbarui Safari ke 18.4<br>  • Konten keamanan Safari 18.4<br><br>Rilis macOS Sonoma 14.7.5<br><br>• Konten keamanan macOS Sonoma 14.7.5<br>• Diperbarui Safari ke 18.4<br>  • Konten keamanan Safari 18.4<br><br>Rilis macOS Sequoia 15.4.1<br><br>• Konten keamanan macOS Sequoia 15.4.1 |

| Rilis | Perubahan |
| --- | --- |
| 2025.03.18 | Semua AMI<br>• Diperbarui `awscli` ke 2.24.2<br>• Memperbarui Homebrew ke 4.4.20<br><br>Rilis macOS Sequoia 15.3.1<br>• Konten keamanan macOS Sequoia 15.3.1<br><br>Rilis macOS Sonoma 14.7.4<br>• Konten keamanan macOS Sonoma 14.7.4<br>• Diperbarui Safari ke 18.3<br><br>Rilis macOS Ventura 13.7.4<br>• Konten keamanan macOS Ventura 13.7.4<br>• Diperbarui Safari ke 18.3 |

| Rilis | Perubahan |
| --- | --- |
| 2025.01.24 | Semua AMI<br>• Diperbarui `awscli` ke 2.22.33<br>• Memperbarui Homebrew ke 4.4.15<br><br>Rilis macOS Sequoia 15.2<br>• Konten keamanan macOS Sequoia 15.2<br>• Alat Baris Perintah yang Diperbarui ke 16.2<br><br>Rilis macOS Sonoma 14.7.2<br>• Konten keamanan macOS Sonoma 14.7.2<br>• Diperbarui Safari ke 18.2<br>• Alat Baris Perintah yang Diperbarui ke 16.2<br><br>Rilis macOS Ventura 13.7.2<br>• Konten keamanan macOS Ventura 13.7.2<br>• Diperbarui Safari ke 18.2 |

| Rilis | Perubahan |
| --- | --- |
| 2024.12.20 | Semua AMI<br>• Memperbarui Homebrew ke 4.4.8<br>• Diperbarui `aws-cli` ke 2.22.5<br>• Diperbarui `amazon-ssm-agent` ke 3.3.987.0<br><br>Rilis macOS Sequoia 15.1.1<br>• Konten keamanan macOS Sequoia 15.1.1<br><br>Rilis macOS Sonoma 14.7.1<br>• Konten keamanan macOS Sonoma 14.7.1<br>• Diperbarui Safari ke 18.1.1<br><br>Rilis macOS Ventura 13.7.1<br>• Konten keamanan macOS Ventura 13.7.1<br>• Diperbarui Safari ke 18.1.1 |

| Rilis | Perubahan |
| --- | --- |
| 2024.10.28 | Semua AMI<br>• Memperbarui Homebrew ke 4.4.2<br>• Diperbarui `aws-cli` ke 2.18.13<br>• Diperbarui `amazon-ssm-agent` ke 3.3.987.0<br>• Diperbarui `ec2-macos-init` ke 1.5.10<br>• Diperbarui `ec2-macos-utils` ke 1.0.4<br><br>Rilis macOS Sequoia 15.0<br>• Konten keamanan macOS Sequoia 15<br><br>Rilis macOS Sonoma 14.7<br>• Konten keamanan macOS Sonoma 14.7.<br>• Alat Baris Perintah yang Diperbarui ke 16.0<br>• Diperbarui Safari ke 18.0.1<br>  • Konten keamanan Safari 18<br><br>Rilis macOS Ventura 13.7<br>• Konten keamanan macOS Ventura 13.7<br>• Diperbarui Safari ke 18.0.1<br>  • Konten keamanan Safari 18 |

| Rilis | Perubahan |
| --- | --- |
| 2024.08.20 | Semua AMI<br>• Memperbarui Homebrew ke 4.3.14<br>• Diperbarui `aws-cli` ke 2.17.29<br><br>Rilis macOS Sonoma 14.6.1<br>• Tidak ada entri CVE yang dipublikasikan.<br><br>Rilis macOS Ventura 13.6.9<br>• Tidak ada entri CVE yang dipublikasikan.<br>• Diperbarui Safari ke 17.6<br>  • Konten keamanan Safari 17.6<br><br>Rilis macOS Monterey 12.7.6<br>• Konten keamanan macOS Monterey 12.7.6<br>• Diperbarui Safari ke 17.6<br>  • Konten keamanan Safari 17.6 |

| Rilis | Perubahan |
| --- | --- |
| 2024.06.07 | Semua AMI<br>• Memperbarui Homebrew ke 4.3.1-1<br>• Diperbarui `aws-cli` ke 2.15.56<br>• Diperbarui `amazon-ssm-agent` ke 3.3.380.0-1<br><br>Rilis macOS Sonoma 14.5<br>• Konten keamanan macOS Sonoma 14.5<br><br>Rilis macOS Ventura 13.6.7<br>• Konten keamanan macOS Ventura 13.6.7<br>• Diperbarui Safari ke 17.5<br>  • Konten keamanan Safari 17.5<br><br>Rilis macOS Monterey 12.7.5<br>• Konten keamanan macOS Monterey 12.7.5<br>• Diperbarui Safari ke 17.5<br>  • Konten keamanan Safari 17.5 |

| Rilis | Perubahan |
| --- | --- |
| 2024.04.12 | Semua AMI<br>• Memperbarui Homebrew ke 4.2.16-1<br>• Diperbarui `aws-cli` ke 2.15.36<br><br>Rilis macOS Sonoma 14.4.1<br>• Konten keamanan macOS Sonoma 14.4.1<br><br>Rilis macOS Ventura 13.6.6<br>• Konten keamanan macOS Ventura 13.6.6<br>• Diperbarui Safari ke 17.4.1<br>  • Konten keamanan Safari 17.4.1<br><br>Untuk macOS Monterey<br>• Diperbarui Safari ke 17.4.1<br>  • Konten keamanan Safari 17.4.1 |

## Jenis EBS-optimized instans Amazon

Instans yang dioptimalkan Amazon EBS menggunakan tumpukan konfigurasi yang dioptimalkan dan menyediakan bandwidth khusus tambahan untuk Amazon EBS. I/O Pengoptimalan ini memberikan kinerja terbaik untuk volume EBS Anda dengan meminimalkan perselisihan antara Amazon EBS I/O dan lalu lintas lain dari instans Anda.

Saat dilampirkan ke instans yang dioptimalkan EBS, volume SSD (gp2dangp3) Tujuan Umum dirancang untuk memberikan setidaknya 90 persen kinerja IOPS yang disediakan 99 persen dari waktu pada tahun tertentu, dan volume SSD (io1danio2) IOPS Tersediakan dirancang untuk memberikan setidaknya 90 persen kinerja IOPS yang disediakan 99,9 persen dari waktu dalam tahun tertentu. Throughput Optimized HDD (st1sc1) dan Cold HDD () memberikan setidaknya 90 persen dari kinerja throughput yang diharapkan 99 persen dari waktu dalam tahun tertentu. Non-compliant periode kira-kira didistribusikan secara seragam, menargetkan 99 persen dari total throughput yang

diharapkan setiap jam. Untuk informasi selengkapnya, lihat Jenis volume Amazon EBS di Panduan Pengguna Amazon EBS.

Beberapa jenis instance secara EBS-optimized default, dan tidak perlu mengaktifkannya dan tidak ada efek jika Anda mencoba menonaktifkannya. Jenis instans lain secara opsional mendukung pengoptimalan EBS dan Anda dapat mengaktifkannya selama atau setelah peluncuran dengan biaya https://aws.amazon.com/ec2/previous-generation/#EBS-optimized_instances tambahan per jam. Beberapa jenis instans tidak mendukung optimasi EBS.

Untuk spesifikasi dan fitur tipe instans terperinci, lihat Panduan Jenis Instans Amazon EC2.

Pertimbangan-pertimbangan

- Kinerja EBS instans dibatasi oleh batas kinerja jenis instans, atau kinerja agregat dari volume terlampirnya, mana saja yang lebih kecil. Untuk mencapai performa EBS maksimum, instans harus memiliki volume terlampir yang memberikan performa gabungan yang sama atau lebih besar dari performa instans maksimum. Misalnya, untuk mencapai `80,000` IOPS untuk`r6i.16xlarge`, instance harus memiliki setidaknya 5 `gp2` volume yang disediakan dengan masing-masing IOPS (5volume x `16,000` IOPS = `16,000` `80,000` IOPS), atau dapat memiliki 1 `gp3` volume yang disediakan dengan IOPS. `80,000` Sebaiknya pilih jenis instans yang menyediakan throughput Amazon EBS yang lebih khusus daripada yang dibutuhkan aplikasi Anda; jika tidak, koneksi antara Amazon EBS dan Amazon EC2 dapat menjadi hambatan kinerja.
- Jumlah maksimum volume Amazon EBS yang dapat dilampirkan ke instans bergantung pada tipe instans dan ukuran instans. Untuk informasi selengkapnya, lihat Batas volume Amazon EBS untuk instans Amazon EC2.
- IOPS maksimum dan batas throughput saling bergantung. Tergantung pada I/O ukuran Anda, Anda mungkin mencapai satu batas sebelum yang lain, yang dapat memengaruhi kinerja keseluruhan. Untuk hasil yang optimal, pertimbangkan kedua batasan saat merencanakan beban kerja Anda.

## EBS-optimized secara default

Jenis instans berikut dioptimalkan EBS secara default. Tidak perlu mengaktifkan optimisasi EBS dan tidak akan ada pengaruh jika Anda menonaktifkan optimasi EBS.

Contoh

- Tujuan umum
- Dioptimalkan untuk komputasi

- Dioptimalkan untuk memori
- Penyimpanan yang dioptimalkan
- Komputasi yang dipercepat
- High-performance komputasi

Tujuan umum

> **Note**
>
> Jenis instans M8a, M8g, M8gd, M8i, M8id,, M9g, M9gd mendukung bobot bandwidth yang dapat dikonfigurasi. M8i-flex Dengan jenis instans ini, Anda dapat mengoptimalkan bandwidth instans untuk kinerja jaringan atau kinerja Amazon EBS. Tabel berikut menunjukkan kinerja bandwidth Amazon EBS default untuk jenis instans ini. Untuk informasi selengkapnya, lihat Konfigurasi pembobotan bandwidth instans EC2.

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| a1.medium [1] | 300 | 3500 | 37.50 | 437.50 | 2500 | 2000 |
| a1.large [1] | 525 | 3500 | 65.62 | 437.50 | 4000 | 2000 |
| a1.xlarge [1] | 800 | 3500 | 100.00 | 437.50 | 6000 | 2000 |
| a1.2xlarge [1] | 1750 | 3500 | 218.75 | 437.50 | 10000 | 2000 |
| a1.4xlarge [2] | 3500 | | 437.5 | | 2000 | |
| a1.metal [2] | 3500 | | 437.5 | | 2000 | |
| m4.large [2] | 450 | | 56.25 | | 3600 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m4.xlarge [2] | 750 | | 93.75 | | 6000 | |
| m4.2xlarge [2] | 1000 | | 125.0 | | 8000 | |
| m4.4xlarge [2] | 2000 | | 250.0 | | 16000 | |
| m4.10xlarge [2] | 4000 | | 500.0 | | 32000 | |
| m4.16xlarge [2] | 10000 | | 1250.0 | | 65000 | |
| m5.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| m5.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |
| m5.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| m5.4xlarge [2] | 4750 | | 593.75 | | 18750 | |
| m5.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| m5.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m5.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |
| m5.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m5.metal [2] | 19000 | | 2375.0 | | 80000 | |
| m5a.large [1] | 650 | 2880 | 81.25 | 360.00 | 3600 | 16000 |
| m5a.xlarge [1] | 1085 | 2880 | 135.62 | 360.00 | 6000 | 16000 |
| m5a.2xlarge [1] | 1580 | 2880 | 197.50 | 360.00 | 8333 | 16000 |
| m5a.4xlarge [2] | 2880 | | 360.0 | | 16000 | |
| m5a.8xlarge [2] | 4750 | | 593.75 | | 2000 | |
| m5a.12xlarge [2] | 6780 | | 847.5 | | 30000 | |
| m5a.16xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m5a.24xlarge [2] | 13750 | | 1718.75 | | 60000 | |
| m5ad.large [1] | 650 | 2880 | 81.25 | 360.00 | 3600 | 16000 |
| m5ad.xlarge [1] | 1085 | 2880 | 135.62 | 360.00 | 6000 | 16000 |
| m5ad.2xlarge [1] | 1580 | 2880 | 197.50 | 360.00 | 8333 | 16000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m5ad.4xlarge [2] | 2880 | | 360.0 | | 16000 | |
| m5ad.8xlarge [2] | 4750 | | 593.75 | | 2000 | |
| m5ad.12xlarge [2] | 6780 | | 847.5 | | 30000 | |
| m5ad.16xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m5ad.24xlarge [2] | 13750 | | 1718.75 | | 60000 | |
| m5d.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| m5d.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |
| m5d.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| m5d.4xlarge [2] | 4750 | | 593.75 | | 18750 | |
| m5d.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| m5d.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m5d.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |
| m5d.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| m5d.metal [2] | 19000 | | 2375.0 | | 80000 | |
| m5dn.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| m5dn.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |
| m5dn.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| m5dn.4xlarge [2] | 4750 | | 593.75 | | 18750 | |
| m5dn.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| m5dn.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m5dn.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |
| m5dn.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m5dn.metal [2] | 19000 | | 2375.0 | | 80000 | |
| m5n.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| m5n.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |
| m5n.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| m5n.4xlarge [2] | 4750 | | 593.75 | | 18750 | |
| m5n.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| m5n.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m5n.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |
| m5n.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| m5n.metal [2] | 19000 | | 2375.0 | | 80000 | |
| m5zn.large [1] | 800 | 3170 | 100.00 | 396.25 | 3333 | 13333 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m5zn.xlarge [1] | 1564 | 3170 | 195.50 | 396.25 | 6667 | 13333 |
| m5zn.2xlarge [2] | 3170 | | 396.25 | | 13333 | |
| m5zn.3xlarge [2] | 4750 | | 593.75 | | 2000 | |
| m5zn.6xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m5zn.12xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| m5zn.metal [2] | 19000 | | 2375.0 | | 80000 | |
| m6a.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m6a.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m6a.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m6a.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m6a.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m6a.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| m6a.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| m6a.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| m6a.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| m6a.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| m6a.metal [2] | 40000 | | 500.0 | | 240000 | |
| m6g.medium [1] | 315 | 4750 | 39.38 | 593.75 | 2500 | 2000 |
| m6g.large [1] | 630 | 4750 | 78.75 | 593.75 | 3600 | 2000 |
| m6g.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |
| m6g.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| m6g.4xlarge [2] | 4750 | | 593.75 | | 2000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m6g.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m6g.12xlarge [2] | 14250 | | 1781.25 | | 5000 | |
| m6g.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| m6g.metal [2] | 19000 | | 2375.0 | | 80000 | |
| m6gd.medium [1] | 315 | 4750 | 39.38 | 593.75 | 2500 | 2000 |
| m6gd.large [1] | 630 | 4750 | 78.75 | 593.75 | 3600 | 2000 |
| m6gd.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |
| m6gd.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| m6gd.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| m6gd.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| m6gd.12xlarge [2] | 14250 | | 1781.25 | | 5000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m6gd.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| m6gd.metal [2] | 19000 | | 2375.0 | | 80000 | |
| m6i.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m6i.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m6i.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m6i.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m6i.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| m6i.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| m6i.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| m6i.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| m6i.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| m6i.metal [2] | 40000 | | 500.0 | | 160000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m6id.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m6id.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m6id.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m6id.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m6id.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| m6id.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| m6id.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| m6id.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| m6id.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| m6id.metal [2] | 40000 | | 500.0 | | 160000 | |
| m6idn.large [1] | 1562 | 25000 | 195.31 | 3125.00 | 6250 | 100000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m6idn.xlarge [1] | 3125 | 25000 | 390.62 | 3125.00 | 12500 | 100000 |
| m6idn.2xlarge [1] | 6250 | 25000 | 781.25 | 3125.00 | 25000 | 100000 |
| m6idn.4xlarge [1] | 12500 | 25000 | 1562.50 | 3125.00 | 5000 | 100000 |
| m6idn.8xlarge [2] | 25000 | | 3125.0 | | 100000 | |
| m6idn.12xlarge [2] | 37500 | | 4687.5 | | 150000 | |
| m6idn.16xlarge [2] | 5000 | | 6250.0 | | 200000 | |
| m6idn.24xlarge [2] | 75000 | | 9375.0 | | 300000 | |
| m6idn.32xlarge [2] | 100000 | | 12500.0 | | 400000 | |
| m6idn.metal [2] | 100000 | | 12500.0 | | 400000 | |
| m6in.large [1] | 1562 | 25000 | 195.31 | 3125.00 | 6250 | 100000 |
| m6in.xlarge [1] | 3125 | 25000 | 390.62 | 3125.00 | 12500 | 100000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m6in.2xlarge [1] | 6250 | 25000 | 781.25 | 3125.00 | 25000 | 100000 |
| m6in.4xlarge [1] | 12500 | 25000 | 1562.50 | 3125.00 | 5000 | 100000 |
| m6in.8xlarge [2] | 25000 | | 3125.0 | | 100000 | |
| m6in.12xlarge [2] | 37500 | | 4687.5 | | 150000 | |
| m6in.16xlarge [2] | 5000 | | 6250.0 | | 200000 | |
| m6in.24xlarge [2] | 75000 | | 9375.0 | | 300000 | |
| m6in.32xlarge [2] | 100000 | | 12500.0 | | 400000 | |
| m6in.metal [2] | 100000 | | 12500.0 | | 400000 | |
| m7a.medium [1] | 325 | 10000 | 40.62 | 1250.00 | 2500 | 40000 |
| m7a.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m7a.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m7a.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m7a.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m7a.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| m7a.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| m7a.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| m7a.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| m7a.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| m7a.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| m7a.metal-48xl [2] | 40000 | | 500.0 | | 240000 | |
| m7g.medium [1] | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| m7g.large [1] | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m7g.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m7g.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m7g.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m7g.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| m7g.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| m7g.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| m7g.metal [2] | 2000 | | 2500.0 | | 80000 | |
| m7gd.medium [1] | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| m7gd.large [1] | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| m7gd.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m7gd.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m7gd.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m7gd.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| m7gd.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| m7gd.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| m7gd.metal 2 | 2000 | | 2500.0 | | 80000 | |
| m7i.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m7i.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m7i.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m7i.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m7i.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| m7i.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| m7i.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m7i.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| m7i.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| m7i.metal-24xl [2] | 30000 | | 3750.0 | | 120000 | |
| m7i.metal-48xl [2] | 40000 | | 500.0 | | 240000 | |
| m7i-flex.large [1] | 312 | 10000 | 39.06 | 1250.00 | 2500 | 40000 |
| m7i-flex.xlarge [1] | 625 | 10000 | 78.12 | 1250.00 | 3600 | 40000 |
| m7i-flex.2xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m7i-flex.4xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m7i-flex.8xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m7i-flex.12xbesar 1 | 7500 | 15000 | 937.50 | 1875.00 | 30000 | 60000 |
| m7i-flex.16xbesar 1 | 10000 | 2000 | 1250.00 | 2500.00 | 40000 | 80000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8a.medium 1 | 325 | 10000 | 40.62 | 1250.00 | 2500 | 40000 |
| m8a.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m8a.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8a.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8a.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8a.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8a.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| m8a.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8a.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| m8a.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| m8a.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8a.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| m8azn.medium 1 | 625 | 15000 | 78.12 | 1875.00 | 2500 | 60000 |
| m8azn.large 1 | 1250 | 15000 | 156.25 | 1875.00 | 5000 | 60000 |
| m8azn.xlarge 1 | 2500 | 15000 | 312.50 | 1875.00 | 10000 | 60000 |
| m8azn.3xbesar 1 | 7500 | 15000 | 937.50 | 1875.00 | 30000 | 60000 |
| m8azn.6xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| m8azn.12xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| m8azn.24xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| m8azn.logam-12xl 2 | 30000 | | 3750.0 | | 120000 | |
| m8azn.logam-24xl 2 | 60000 | | 7500.0 | | 240000 | |
| m8g.sedang 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8g.besar 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| m8g.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8g.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8g.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8g.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8g.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| m8g.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8g.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| m8g.48xbesar 2 | 40000 | | 500.0 | | 240000 | |
| m8g.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| m8g.logam-48xl 2 | 40000 | | 500.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8gb.medium 1 | 1562 | 25000 | 195.31 | 3125.00 | 7500 | 120000 |
| m8gb.besar 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| m8gb.xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| m8gb.2xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |
| m8gb.4xbesar 2 | 25000 | | 3125.0 | | 120000 | |
| m8gb.8xbesar 2 | 5000 | | 6250.0 | | 240000 | |
| m8gb.12xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| m8gb.16xbesar 2 | 100000 | | 12500.0 | | 480000 | |
| m8gb.24xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| m8gb.48xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| m8gb.logam-24xl 2 | 150000 | | 18750.0 | | 72000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8gb.logam-48xl 2 | 300000 | | 37500.0 | | 1440000 | |
| m8gd.medium 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| m8gd.besar 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| m8gd.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8gd.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8gd.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8gd.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8gd.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| m8gd.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8gd.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| m8gd.48xbesar 2 | 40000 | | 500.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8gd.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| m8gd.logam-48xl 2 | 40000 | | 500.0 | | 240000 | |
| m8gn.sedang 1 | 760 | 10000 | 95.00 | 1250.00 | 2500 | 40000 |
| m8gn.besar 1 | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| m8gn.xlarge 1 | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| m8gn.2xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8gn.4xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8gn.8xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8gn.12xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| m8gn.16xbesar 2 | 40000 | | 500.0 | | 160000 | |
| m8gn.24xbesar 2 | 60000 | | 7500.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8gn.48xbesar 2 | 120000 | | 1500.0 | | 480000 | |
| m8gn.logam-24xl 2 | 60000 | | 7500.0 | | 240000 | |
| m8gn.logam-48xl 2 | 120000 | | 1500.0 | | 480000 | |
| m8i.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m8i.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8i.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8i.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8i.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8i.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| m8i.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8i.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8i.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| m8i.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| m8i.96xbesar 2 | 80000 | | 1000.0 | | 480000 | |
| m8i.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| m8i.logam-96xl 2 | 80000 | | 1000.0 | | 480000 | |
| m8id.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m8id.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8id.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8id.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8id.8xbesar 2 | 1000 | | 1250.0 | | 40000 | |
| m8id.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8id.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8id.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| m8id.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| m8id.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| m8id.96xbesar 2 | 80000 | | 1000.0 | | 480000 | |
| m8id.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| m8id.logam-96xl 2 | 80000 | | 1000.0 | | 480000 | |
| m8i-flex.besar 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| m8i-flex.xlarge 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| m8i-flex.2xbesar 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8i-flex.4xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8i-flex.8xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8i-flex.12xbesar 1 | 7500 | 15000 | 937.50 | 1875.00 | 30000 | 60000 |
| m8i-flex.16xbesar 1 | 10000 | 2000 | 1250.00 | 2500.00 | 40000 | 80000 |
| m8in.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m8in.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8in.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8in.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8in.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8in.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| m8in.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8in.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8in.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| m8in.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| m8in.96xbesar 2 | 120000 | | 1500.0 | | 480000 | |
| m8in.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| m8in.logam-96xl 2 | 120000 | | 1500.0 | | 480000 | |
| m8idn.large 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m8idn.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8idn.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8idn.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| m8idn.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8idn.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8idn.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| m8idn.24xlarge 2 | 30000 | | 3750.0 | | 120000 | |
| m8idn.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| m8idn.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| m8idn.96xbesar 2 | 120000 | | 1500.0 | | 480000 | |
| m8idn.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| m8idn.logam-96xl 2 | 120000 | | 1500.0 | | 480000 | |
| m8ine.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| m8ine.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| m8ine.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| m8ine.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8ine.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| m8ine.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| m8ib.besar 1 | 1563 | 25000 | 195.38 | 3125.00 | 7500 | 120000 |
| m8ib.xbesar 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| m8ib.2xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| m8ib.4xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |
| m8ib.8xbesar 2 | 25000 | | 3125.0 | | 120000 | |
| m8ib.12xbesar 2 | 37500 | | 4687.5 | | 180000 | |
| m8ib.16xbesar 2 | 5000 | | 6250.0 | | 240000 | |
| m8ib.24xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| m8ib.32xbesar 2 | 100000 | | 12500.0 | | 480000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8ib.48xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| m8ib.96xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| m8ib.logam-48xl 2 | 150000 | | 18750.0 | | 72000 | |
| m8ib.logam-96xl 2 | 300000 | | 37500.0 | | 1440000 | |
| m8idb.besar 1 | 1563 | 25000 | 195.38 | 3125.00 | 7500 | 120000 |
| m8idb.xlarge 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| m8idb.2xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| m8idb.4xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |
| m8idb.8xbesar 2 | 25000 | | 3125.0 | | 120000 | |
| m8idb.12xbesar 2 | 37500 | | 4687.5 | | 180000 | |
| m8idb.16xbesar 2 | 5000 | | 6250.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m8idb.24xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| m8idb.32xbesar 2 | 100000 | | 12500.0 | | 480000 | |
| m8idb.48xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| m8idb.96xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| m8idb.logam-48xl 2 | 150000 | | 18750.0 | | 72000 | |
| m8idb.logam-96xl 2 | 300000 | | 37500.0 | | 1440000 | |
| m9g.sedang 1 | 380 | 12000 | 47.50 | 150.00 | 2500 | 48000 |
| m9g.besar 1 | 760 | 12000 | 95.00 | 150.00 | 3600 | 48000 |
| m9g.xlarge 1 | 1500 | 12000 | 187.50 | 150.00 | 6000 | 48000 |
| m9g.2xbesar 1 | 3000 | 12000 | 375.00 | 150.00 | 12000 | 48000 |
| m9g.4xbesar 1 | 6000 | 12000 | 750.00 | 150.00 | 24000 | 48000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m9g.8xbesar 2 | 12000 | | 150.0 | | 48000 | |
| m9g.12xbesar 2 | 18000 | | 2250.0 | | 72000 | |
| m9g.16xbesar 2 | 24000 | | 3000.0 | | 96000 | |
| m9g.24xbesar 2 | 36000 | | 4500.0 | | 144000 | |
| m9g.48xbesar 2 | 72000 | | 900.0 | | 288000 | |
| m9g.logam-48xl 2 | 72000 | | 900.0 | | 288000 | |
| m9gd.medium 1 | 380 | 12000 | 47.50 | 150.00 | 2500 | 48000 |
| m9gd.besar 1 | 760 | 12000 | 95.00 | 150.00 | 3600 | 48000 |
| m9gd.xlarge 1 | 1500 | 12000 | 187.50 | 150.00 | 6000 | 48000 |
| m9gd.2xbesar 1 | 3000 | 12000 | 375.00 | 150.00 | 12000 | 48000 |
| m9gd.4xbesar 1 | 6000 | 12000 | 750.00 | 150.00 | 24000 | 48000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| m9gd.8xbesar 2 | 12000 | | 150.0 | | 48000 | |
| m9gd.12xbesar 2 | 18000 | | 2250.0 | | 72000 | |
| m9gd.16xbesar 2 | 24000 | | 3000.0 | | 96000 | |
| m9gd.24xbesar 2 | 36000 | | 4500.0 | | 144000 | |
| m9gd.48xbesar 2 | 72000 | | 900.0 | | 288000 | |
| m9gd.logam-48xl 2 | 72000 | | 900.0 | | 288000 | |
| mac1.metal [2] | 14000 | | 1750.0 | | 80000 | |
| mac2.metal [2] | 10000 | | 1250.0 | | 55000 | |
| mac2-m1ultra metal 2 | 10000 | | 1250.0 | | 55000 | |
| mac2-m2.metal [2] | 8000 | | 1000.0 | | 55000 | |
| mac2-m2pro.metal [2] | 8000 | | 1000.0 | | 55000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| mac-m3ultra.metal 2 | 10000 | | 1250.0 | | 55000 | |
| mac-m4.metal 2 | 8000 | | 1000.0 | | 55000 | |
| mac-m4pro.metal 2 | 8000 | | 1000.0 | | 55000 | |
| mac-m4max.metal 2 | 10000 | | 1250.0 | | 55000 | |
| t3.nano [1] | 43 | 2085 | 5.38 | 260.62 | 250 | 11800 |
| t3.micro [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11800 |
| t3.small [1] | 174 | 2085 | 21.75 | 260.62 | 1000 | 11800 |
| t3.medium [1] | 347 | 2085 | 43.38 | 260.62 | 2000 | 11800 |
| t3.large [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t3.xlarge [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t3.2xlarge [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t3a.nano [1] | 45 | 2085 | 5.62 | 260.62 | 250 | 11800 |
| t3a.micro [1] | 90 | 2085 | 11.25 | 260.62 | 500 | 11800 |
| t3a.small [1] | 175 | 2085 | 21.88 | 260.62 | 1000 | 11800 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| t3a.medium [1] | 350 | 2085 | 43.75 | 260.62 | 2000 | 11800 |
| t3a.large [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t3a.xlarge [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t3a.2xlarge [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t4g.nano [1] | 43 | 2085 | 5.38 | 260.62 | 250 | 11800 |
| t4g.micro [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11800 |
| t4g.small [1] | 174 | 2085 | 21.75 | 260.62 | 1000 | 11800 |
| t4g.medium [1] | 347 | 2085 | 43.38 | 260.62 | 2000 | 11800 |
| t4g.large [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t4g.xlarge [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |
| t4g.2xlarge [1] | 695 | 2780 | 86.88 | 347.50 | 4000 | 15700 |

[1] Instans ini dapat mempertahankan kinerja maksimum selama 30 menit setidaknya sekali setiap 24 jam, setelah itu mereka kembali ke kinerja dasar.

[2] Instans ini dapat mempertahankan performa yang dinyatakan tanpa batas waktu. Jika beban kerja Anda memerlukan performa maksimum yang berkelanjutan selama lebih dari 30 menit, gunakan salah satu instans ini.

Dioptimalkan untuk komputasi

> **Note**
>
> Jenis instance C8a, C8g, C8gd, C8i, C8id,, C9g, C9gd mendukung bobot bandwidth yang dapat dikonfigurasi. C8i-flex Dengan jenis instans ini, Anda dapat mengoptimalkan bandwidth instans untuk kinerja jaringan atau kinerja Amazon EBS. Tabel berikut menunjukkan kinerja bandwidth Amazon EBS default untuk jenis instans ini. Untuk informasi selengkapnya, lihat Konfigurasi pembobotan bandwidth instans EC2.

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c4.large [2] | 500 | | 62.5 | | 4000 | |
| c4.xlarge [2] | 750 | | 93.75 | | 6000 | |
| c4.2xlarge [2] | 1000 | | 125.0 | | 8000 | |
| c4.4xlarge [2] | 2000 | | 250.0 | | 16000 | |
| c4.8xlarge [2] | 4000 | | 500.0 | | 32000 | |
| c5.large [1] | 650 | 4750 | 81.25 | 593.75 | 4000 | 2000 |
| c5.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 2000 |
| c5.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 10000 | 2000 |
| c5.4xlarge [2] | 4750 | | 593.75 | | 2000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c5.9xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c5.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c5.18xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c5.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c5.metal [2] | 19000 | | 2375.0 | | 80000 | |
| c5a.large [1] | 200 | 3170 | 25.00 | 396.25 | 800 | 13300 |
| c5a.xlarge [1] | 400 | 3170 | 50.00 | 396.25 | 1600 | 13300 |
| c5a.2xlarge [1] | 800 | 3170 | 100.00 | 396.25 | 3200 | 13300 |
| c5a.4xlarge [1] | 1580 | 3170 | 197.50 | 396.25 | 6600 | 13300 |
| c5a.8xlarge [2] | 3170 | | 396.25 | | 13300 | |
| c5a.12xlarge [2] | 4750 | | 593.75 | | 2000 | |
| c5a.16xlarge [2] | 6300 | | 787.5 | | 26700 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c5a.24xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c5ad.large [1] | 200 | 3170 | 25.00 | 396.25 | 800 | 13300 |
| c5ad.xlarge [1] | 400 | 3170 | 50.00 | 396.25 | 1600 | 13300 |
| c5ad.2xlarge [1] | 800 | 3170 | 100.00 | 396.25 | 3200 | 13300 |
| c5ad.4xlarge [1] | 1580 | 3170 | 197.50 | 396.25 | 6600 | 13300 |
| c5ad.8xlarge [2] | 3170 | | 396.25 | | 13300 | |
| c5ad.12xlarge [2] | 4750 | | 593.75 | | 2000 | |
| c5ad.16xlarge [2] | 6300 | | 787.5 | | 26700 | |
| c5ad.24xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c5d.large [1] | 650 | 4750 | 81.25 | 593.75 | 4000 | 2000 |
| c5d.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 2000 |
| c5d.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 10000 | 2000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c5d.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| c5d.9xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c5d.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c5d.18xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c5d.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c5d.metal [2] | 19000 | | 2375.0 | | 80000 | |
| c5n.large [1] | 650 | 4750 | 81.25 | 593.75 | 4000 | 2000 |
| c5n.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 2000 |
| c5n.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 10000 | 2000 |
| c5n.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| c5n.9xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c5n.18xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c5n.metal [2] | 19000 | | 2375.0 | | 80000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c6a.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c6a.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c6a.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c6a.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c6a.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| c6a.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| c6a.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| c6a.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| c6a.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| c6a.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| c6a.metal [2] | 40000 | | 500.0 | | 240000 | |
| c6g.medium [1] | 315 | 4750 | 39.38 | 593.75 | 2500 | 2000 |
| c6g.large [1] | 630 | 4750 | 78.75 | 593.75 | 3600 | 2000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c6g.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |
| c6g.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| c6g.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| c6g.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c6g.12xlarge [2] | 14250 | | 1781.25 | | 5000 | |
| c6g.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c6g.metal [2] | 19000 | | 2375.0 | | 80000 | |
| c6gd.medium [1] | 315 | 4750 | 39.38 | 593.75 | 2500 | 2000 |
| c6gd.large [1] | 630 | 4750 | 78.75 | 593.75 | 3600 | 2000 |
| c6gd.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |
| c6gd.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| c6gd.4xlarge [2] | 4750 | | 593.75 | | 2000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c6gd.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c6gd.12xlarge [2] | 14250 | | 1781.25 | | 5000 | |
| c6gd.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c6gd.metal [2] | 19000 | | 2375.0 | | 80000 | |
| c6gn.medium [1] | 760 | 9500 | 95.00 | 1187.50 | 2500 | 40000 |
| c6gn.large [1] | 1235 | 9500 | 154.38 | 1187.50 | 5000 | 40000 |
| c6gn.xlarge [1] | 2375 | 9500 | 296.88 | 1187.50 | 10000 | 40000 |
| c6gn.2xlarge [1] | 4750 | 9500 | 593.75 | 1187.50 | 2000 | 40000 |
| c6gn.4xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| c6gn.8xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| c6gn.12xlarge [2] | 28500 | | 3562.5 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c6gn.16xlarge [2] | 38000 | | 4750.0 | | 160000 | |
| c6i.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c6i.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c6i.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c6i.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c6i.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| c6i.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| c6i.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| c6i.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| c6i.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| c6i.metal [2] | 40000 | | 500.0 | | 160000 | |
| c6id.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c6id.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c6id.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c6id.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c6id.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| c6id.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| c6id.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| c6id.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| c6id.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| c6id.metal [2] | 40000 | | 500.0 | | 160000 | |
| c6in.large [1] | 1562 | 25000 | 195.31 | 3125.00 | 6250 | 100000 |
| c6in.xlarge [1] | 3125 | 25000 | 390.62 | 3125.00 | 12500 | 100000 |
| c6in.2xlarge [1] | 6250 | 25000 | 781.25 | 3125.00 | 25000 | 100000 |
| c6in.4xlarge [1] | 12500 | 25000 | 1562.50 | 3125.00 | 5000 | 100000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c6in.8xlarge [2] | 25000 | | 3125.0 | | 100000 | |
| c6in.12xlarge [2] | 37500 | | 4687.5 | | 150000 | |
| c6in.16xlarge [2] | 5000 | | 6250.0 | | 200000 | |
| c6in.24xlarge [2] | 75000 | | 9375.0 | | 300000 | |
| c6in.32xlarge [2] | 100000 | | 12500.0 | | 400000 | |
| c6in.metal [2] | 100000 | | 12500.0 | | 400000 | |
| c7a.medium [1] | 325 | 10000 | 40.62 | 1250.00 | 2500 | 40000 |
| c7a.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c7a.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c7a.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c7a.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c7a.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c7a.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| c7a.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| c7a.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| c7a.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| c7a.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| c7a.metal-48xl [2] | 40000 | | 500.0 | | 240000 | |
| c7g.medium [1] | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| c7g.large [1] | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| c7g.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c7g.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c7g.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c7g.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c7g.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| c7g.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| c7g.metal [2] | 2000 | | 2500.0 | | 80000 | |
| c7gd.medium [1] | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| c7gd.large [1] | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| c7gd.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c7gd.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c7gd.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c7gd.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| c7gd.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| c7gd.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| c7gd.metal 2 | 2000 | | 2500.0 | | 80000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c7gn.medium [1] | 521 | 10000 | 65.12 | 1250.00 | 2083 | 40000 |
| c7gn.large [1] | 1042 | 10000 | 130.25 | 1250.00 | 4167 | 40000 |
| c7gn.xlarge [1] | 2083 | 10000 | 260.38 | 1250.00 | 8333 | 40000 |
| c7gn.2xlarge [1] | 4167 | 10000 | 520.88 | 1250.00 | 16667 | 40000 |
| c7gn.4xlarge [1] | 8333 | 10000 | 1041.62 | 1250.00 | 33333 | 40000 |
| c7gn.8xlarge [1] | 16667 | 2000 | 2083.38 | 2500.00 | 66667 | 80000 |
| c7gn.12xlarge [1] | 25000 | 30000 | 3125.00 | 3750.00 | 100000 | 120000 |
| c7gn.16xlarge [1] | 33333 | 40000 | 4166.62 | 5000.00 | 133333 | 160000 |
| c7gn.logam 1 | 33333 | 40000 | 4166.62 | 5000.00 | 133333 | 160000 |
| c7i.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c7i.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c7i.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c7i.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c7i.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| c7i.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| c7i.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| c7i.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| c7i.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| c7i.metal-24xl [2] | 30000 | | 3750.0 | | 120000 | |
| c7i.metal-48xl [2] | 40000 | | 500.0 | | 240000 | |
| c7i-flex.besar 1 | 312 | 10000 | 39.06 | 1250.00 | 2500 | 40000 |
| c7i-flex.xlarge 1 | 625 | 10000 | 78.12 | 1250.00 | 3600 | 40000 |
| c7i-flex.2xbesar 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c7i-flex.4xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c7i-flex.8xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c7i-flex.12xbesar 1 | 7500 | 15000 | 937.50 | 1875.00 | 30000 | 60000 |
| c7i-flex.16xbesar 1 | 10000 | 2000 | 1250.00 | 2500.00 | 40000 | 80000 |
| c8a.medium 1 | 325 | 10000 | 40.62 | 1250.00 | 2500 | 40000 |
| c8a.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c8a.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c8a.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c8a.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8a.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| c8a.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8a.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| c8a.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| c8a.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| c8a.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| c8a.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| c8g.sedang 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| c8g.besar 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| c8g.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c8g.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c8g.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8g.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8g.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| c8g.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| c8g.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| c8g.48xbesar 2 | 40000 | | 500.0 | | 240000 | |
| c8g.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| c8g.logam-48xl 2 | 40000 | | 500.0 | | 240000 | |
| c8gb.sedang 1 | 1562 | 25000 | 195.31 | 3125.00 | 7500 | 120000 |
| c8gb.besar 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| c8gb.xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| c8gb.2xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |
| c8gb.4xbesar 2 | 25000 | | 3125.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8gb.8xbesar 2 | 5000 | | 6250.0 | | 240000 | |
| c8gb.12xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| c8gb.16xbesar 2 | 100000 | | 12500.0 | | 480000 | |
| c8gb.24xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| c8gb.48xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| c8gb.logam-24xl 2 | 150000 | | 18750.0 | | 72000 | |
| c8gb.logam-48xl 2 | 300000 | | 37500.0 | | 1440000 | |
| c8gd.medium 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| c8gd.large 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| c8gd.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c8gd.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8gd.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8gd.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| c8gd.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| c8gd.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| c8gd.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| c8gd.48xbesar 2 | 40000 | | 500.0 | | 240000 | |
| c8gd.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| c8gd.logam-48xl 2 | 40000 | | 500.0 | | 240000 | |
| c8gn.sedang 1 | 760 | 10000 | 95.00 | 1250.00 | 2500 | 40000 |
| c8gn.besar 1 | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| c8gn.xlarge 1 | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8gn.2xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8gn.4xbesar 2 | 1000 | | 1250.0 | | 40000 | |
| c8gn.8xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| c8gn.12xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| c8gn.16xbesar 2 | 40000 | | 500.0 | | 160000 | |
| c8gn.24xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| c8gn.48xbesar 2 | 120000 | | 1500.0 | | 480000 | |
| c8gn.logam-24xl 2 | 60000 | | 7500.0 | | 240000 | |
| c8gn.logam-48xl 2 | 120000 | | 1500.0 | | 480000 | |
| c8i.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c8i.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8i.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c8i.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8i.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| c8i.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| c8i.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| c8i.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| c8i.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| c8i.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| c8i.96xbesar 2 | 80000 | | 1000.0 | | 480000 | |
| c8i.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| c8i.logam-96xl 2 | 80000 | | 1000.0 | | 480000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8id.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c8id.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c8id.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c8id.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8id.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| c8id.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| c8id.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| c8id.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| c8id.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| c8id.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| c8id.96xbesar 2 | 80000 | | 1000.0 | | 480000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8id.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| c8id.logam-96xl 2 | 80000 | | 1000.0 | | 480000 | |
| c8i-flex.besar 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| c8i-flex.xlarge 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| c8i-flex.2xbesar 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c8i-flex.4xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c8i-flex.8xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8i-flex.12xbesar 1 | 7500 | 15000 | 937.50 | 1875.00 | 30000 | 60000 |
| c8i-flex.16xbesar 1 | 1000 | 2000 | 1250.00 | 2500.00 | 40000 | 80000 |
| c8in.large 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c8in.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8in.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c8in.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8in.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| c8in.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| c8in.16xlarge 2 | 2000 | | 2500.0 | | 80000 | |
| c8in.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| c8in.32xlarge 2 | 40000 | | 500.0 | | 160000 | |
| c8in.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| c8in.96xlarge 2 | 120000 | | 1500.0 | | 480000 | |
| c8in.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| c8in.logam-96xl 2 | 120000 | | 1500.0 | | 480000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8ine.large 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| c8ine.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| c8ine.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| c8ine.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| c8ine.8xbesar 2 | 1000 | | 1250.0 | | 40000 | |
| c8ine.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| c8ib.besar 1 | 1563 | 25000 | 195.38 | 3125.00 | 7500 | 120000 |
| c8ib.xlarge 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| c8ib.2xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| c8ib.4xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |
| c8ib.8xbesar 2 | 25000 | | 3125.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c8ib.12xbesar 2 | 37500 | | 4687.5 | | 180000 | |
| c8ib.16xbesar 2 | 5000 | | 6250.0 | | 240000 | |
| c8ib.24xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| c8ib.32xbesar 2 | 100000 | | 12500.0 | | 480000 | |
| c8ib.48xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| c8ib.96xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| c8ib.logam-48xl 2 | 150000 | | 18750.0 | | 72000 | |
| c8ib.logam-96xl 2 | 300000 | | 37500.0 | | 1440000 | |
| c9g.sedang 1 | 380 | 12000 | 47.50 | 150.00 | 2500 | 48000 |
| c9g.besar 1 | 760 | 12000 | 95.00 | 150.00 | 3600 | 48000 |
| c9g.xlarge 1 | 1500 | 12000 | 187.50 | 150.00 | 6000 | 48000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c9g.2xbesar 1 | 3000 | 12000 | 375.00 | 150.00 | 12000 | 48000 |
| c9g.4xbesar 1 | 6000 | 12000 | 750.00 | 150.00 | 24000 | 48000 |
| c9g.8xbesar 2 | 12000 | | 150.0 | | 48000 | |
| c9g.12xbesar 2 | 18000 | | 2250.0 | | 72000 | |
| c9g.16xbesar 2 | 24000 | | 3000.0 | | 96000 | |
| c9g.24xbesar 2 | 36000 | | 4500.0 | | 144000 | |
| c9g.48xbesar 2 | 72000 | | 900.0 | | 288000 | |
| c9g.logam-48xl 2 | 72000 | | 900.0 | | 288000 | |
| c9gd.medium 1 | 380 | 12000 | 47.50 | 150.00 | 2500 | 48000 |
| c9gd.large 1 | 760 | 12000 | 95.00 | 150.00 | 3600 | 48000 |
| c9gd.xlarge 1 | 1500 | 12000 | 187.50 | 150.00 | 6000 | 48000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| c9gd.2xbesar 1 | 3000 | 12000 | 375.00 | 150.00 | 12000 | 48000 |
| c9gd.4xbesar 1 | 6000 | 12000 | 750.00 | 150.00 | 24000 | 48000 |
| c9gd.8xbesar 2 | 12000 | | 150.0 | | 48000 | |
| c9gd.12xbesar 2 | 18000 | | 2250.0 | | 72000 | |
| c9gd.16xbesar 2 | 24000 | | 3000.0 | | 96000 | |
| c9gd.24xbesar 2 | 36000 | | 4500.0 | | 144000 | |
| c9gd.48xbesar 2 | 72000 | | 900.0 | | 288000 | |
| c9gd.logam-48xl 2 | 72000 | | 900.0 | | 288000 | |

[1] Instans ini dapat mempertahankan kinerja maksimum selama 30 menit setidaknya sekali setiap 24 jam, setelah itu mereka kembali ke kinerja dasar.

[2] Instans ini dapat mempertahankan performa yang dinyatakan tanpa batas waktu. Jika beban kerja Anda memerlukan performa maksimum yang berkelanjutan selama lebih dari 30 menit, gunakan salah satu instans ini.

Dioptimalkan untuk memori

> **Note**
>
> - Jenis instance R8a, R8g, R8gd, R8i, R8id,, R9g, R9gd, X8g, X8aedz R8i-flex, X8i mendukung bobot bandwidth yang dapat dikonfigurasi. Dengan jenis instans ini, Anda dapat mengoptimalkan bandwidth instans untuk kinerja jaringan atau kinerja Amazon EBS. Tabel berikut menunjukkan kinerja bandwidth Amazon EBS default untuk jenis instans ini. Untuk informasi selengkapnya, lihat Konfigurasi pembobotan bandwidth instans EC2.
> - Untuk kinerja IOPS maksimum dengan instans U7i, sebaiknya gunakan volume io2. BlockExpress

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r4.large [2] | 425 | | 53.125 | | 3000 | |
| r4.xlarge [2] | 850 | | 106.25 | | 6000 | |
| r4.2xlarge [2] | 1700 | | 212.5 | | 12000 | |
| r4.4xlarge [2] | 3500 | | 437.5 | | 18750 | |
| r4.8xlarge [2] | 7000 | | 875.0 | | 37500 | |
| r4.16xlarge [2] | 14000 | | 1750.0 | | 75000 | |
| r5.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| r5.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r5.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| r5.4xlarge [2] | 4750 | | 593.75 | | 18750 | |
| r5.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| r5.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| r5.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |
| r5.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| r5.metal [2] | 19000 | | 2375.0 | | 80000 | |
| r5a.large [1] | 650 | 2880 | 81.25 | 360.00 | 3600 | 16000 |
| r5a.xlarge [1] | 1085 | 2880 | 135.62 | 360.00 | 6000 | 16000 |
| r5a.2xlarge [1] | 1580 | 2880 | 197.50 | 360.00 | 8333 | 16000 |
| r5a.4xlarge [2] | 2880 | | 360.0 | | 16000 | |
| r5a.8xlarge [2] | 4750 | | 593.75 | | 2000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r5a.12xlarge [2] | 6780 | | 847.5 | | 30000 | |
| r5a.16xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| r5a.24xlarge [2] | 13570 | | 1696.25 | | 60000 | |
| r5ad.large [1] | 650 | 2880 | 81.25 | 360.00 | 3600 | 16000 |
| r5ad.xlarge [1] | 1085 | 2880 | 135.62 | 360.00 | 6000 | 16000 |
| r5ad.2xlarge [1] | 1580 | 2880 | 197.50 | 360.00 | 8333 | 16000 |
| r5ad.4xlarge [2] | 2880 | | 360.0 | | 16000 | |
| r5ad.8xlarge [2] | 4750 | | 593.75 | | 2000 | |
| r5ad.12xlarge [2] | 6780 | | 847.5 | | 30000 | |
| r5ad.16xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| r5ad.24xlarge [2] | 13570 | | 1696.25 | | 60000 | |
| r5b.large [1] | 1250 | 10000 | 156.25 | 1250.00 | 5417 | 43333 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r5b.xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 10833 | 43333 |
| r5b.2xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 21667 | 43333 |
| r5b.4xlarge [2] | 10000 | | 1250.0 | | 43333 | |
| r5b.8xlarge [2] | 2000 | | 2500.0 | | 86667 | |
| r5b.12xlarge [2] | 30000 | | 3750.0 | | 130000 | |
| r5b.16xlarge [2] | 40000 | | 500.0 | | 173333 | |
| r5b.24xlarge [2] | 60000 | | 7500.0 | | 260000 | |
| r5b.metal [2] | 60000 | | 7500.0 | | 260000 | |
| r5d.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| r5d.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |
| r5d.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| r5d.4xlarge [2] | 4750 | | 593.75 | | 18750 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r5d.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| r5d.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| r5d.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |
| r5d.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| r5d.metal [2] | 19000 | | 2375.0 | | 80000 | |
| r5dn.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| r5dn.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |
| r5dn.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| r5dn.4xlarge [2] | 4750 | | 593.75 | | 18750 | |
| r5dn.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| r5dn.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| r5dn.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r5dn.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| r5dn.metal [2] | 19000 | | 2375.0 | | 80000 | |
| r5n.large [1] | 650 | 4750 | 81.25 | 593.75 | 3600 | 18750 |
| r5n.xlarge [1] | 1150 | 4750 | 143.75 | 593.75 | 6000 | 18750 |
| r5n.2xlarge [1] | 2300 | 4750 | 287.50 | 593.75 | 12000 | 18750 |
| r5n.4xlarge [2] | 4750 | | 593.75 | | 18750 | |
| r5n.8xlarge [2] | 6800 | | 850.0 | | 30000 | |
| r5n.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| r5n.16xlarge [2] | 13600 | | 1700.0 | | 60000 | |
| r5n.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| r5n.metal [2] | 19000 | | 2375.0 | | 80000 | |
| r6a.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r6a.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r6a.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r6a.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r6a.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| r6a.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| r6a.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r6a.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| r6a.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| r6a.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| r6a.metal [2] | 40000 | | 500.0 | | 240000 | |
| r6g.medium [1] | 315 | 4750 | 39.38 | 593.75 | 2500 | 2000 |
| r6g.large [1] | 630 | 4750 | 78.75 | 593.75 | 3600 | 2000 |
| r6g.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r6g.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| r6g.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| r6g.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| r6g.12xlarge [2] | 14250 | | 1781.25 | | 5000 | |
| r6g.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| r6g.metal [2] | 19000 | | 2375.0 | | 80000 | |
| r6gd.medium [1] | 315 | 4750 | 39.38 | 593.75 | 2500 | 2000 |
| r6gd.large [1] | 630 | 4750 | 78.75 | 593.75 | 3600 | 2000 |
| r6gd.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |
| r6gd.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| r6gd.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| r6gd.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r6gd.12xlarge [2] | 14250 | | 1781.25 | | 5000 | |
| r6gd.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| r6gd.metal [2] | 19000 | | 2375.0 | | 80000 | |
| r6i.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r6i.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r6i.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r6i.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r6i.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| r6i.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| r6i.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r6i.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| r6i.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| r6i.metal [2] | 40000 | | 500.0 | | 160000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r6id.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r6id.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r6id.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r6id.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r6id.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| r6id.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| r6id.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r6id.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| r6id.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| r6id.metal [2] | 40000 | | 500.0 | | 160000 | |
| r6idn.large [1] | 1562 | 25000 | 195.31 | 3125.00 | 6250 | 100000 |
| r6idn.xlarge [1] | 3125 | 25000 | 390.62 | 3125.00 | 12500 | 100000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r6idn.2xlarge [1] | 6250 | 25000 | 781.25 | 3125.00 | 25000 | 100000 |
| r6idn.4xlarge [1] | 12500 | 25000 | 1562.50 | 3125.00 | 5000 | 100000 |
| r6idn.8xlarge [2] | 25000 | | 3125.0 | | 100000 | |
| r6idn.12xlarge [2] | 37500 | | 4687.5 | | 150000 | |
| r6idn.16xlarge [2] | 5000 | | 6250.0 | | 200000 | |
| r6idn.24xlarge [2] | 75000 | | 9375.0 | | 300000 | |
| r6idn.32xlarge [2] | 100000 | | 12500.0 | | 400000 | |
| r6idn.metal [2] | 100000 | | 12500.0 | | 400000 | |
| r6in.large [1] | 1562 | 25000 | 195.31 | 3125.00 | 6250 | 100000 |
| r6in.xlarge [1] | 3125 | 25000 | 390.62 | 3125.00 | 12500 | 100000 |
| r6in.2xlarge [1] | 6250 | 25000 | 781.25 | 3125.00 | 25000 | 100000 |
| r6in.4xlarge [1] | 12500 | 25000 | 1562.50 | 3125.00 | 5000 | 100000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r6in.8xlarge [2] | 25000 | | 3125.0 | | 100000 | |
| r6in.12xlarge [2] | 37500 | | 4687.5 | | 150000 | |
| r6in.16xlarge [2] | 5000 | | 6250.0 | | 200000 | |
| r6in.24xlarge [2] | 75000 | | 9375.0 | | 300000 | |
| r6in.32xlarge [2] | 100000 | | 12500.0 | | 400000 | |
| r6in.metal [2] | 100000 | | 12500.0 | | 400000 | |
| r7a.medium [1] | 325 | 10000 | 40.62 | 1250.00 | 2500 | 40000 |
| r7a.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r7a.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r7a.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r7a.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r7a.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r7a.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| r7a.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r7a.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| r7a.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| r7a.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| r7a.metal-48xl [2] | 40000 | | 500.0 | | 240000 | |
| r7g.medium [1] | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| r7g.large [1] | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| r7g.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r7g.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r7g.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r7g.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r7g.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| r7g.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r7g.metal [2] | 2000 | | 2500.0 | | 80000 | |
| r7gd.medium [1] | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| r7gd.large [1] | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| r7gd.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r7gd.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r7gd.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r7gd.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| r7gd.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| r7gd.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r7gd.metal 2 | 2000 | | 2500.0 | | 80000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r7i.large [1] | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r7i.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r7i.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r7i.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r7i.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| r7i.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| r7i.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r7i.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| r7i.48xlarge [2] | 40000 | | 500.0 | | 240000 | |
| r7i.metal-24xl [2] | 30000 | | 3750.0 | | 120000 | |
| r7i.metal-48xl [2] | 40000 | | 500.0 | | 240000 | |
| r7iz.large [1] | 792 | 10000 | 99.00 | 1250.00 | 3600 | 40000 |
| r7iz.xlarge [1] | 1584 | 10000 | 198.00 | 1250.00 | 6667 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r7iz.2xlarge [1] | 3168 | 10000 | 396.00 | 1250.00 | 13333 | 40000 |
| r7iz.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r7iz.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| r7iz.12xlarge [2] | 19000 | | 2375.0 | | 76000 | |
| r7iz.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| r7iz.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| r7iz.metal-16xl [2] | 2000 | | 2500.0 | | 80000 | |
| r7iz.metal-32xl [2] | 40000 | | 500.0 | | 160000 | |
| r8a.medium 1 | 325 | 10000 | 40.62 | 1250.00 | 2500 | 40000 |
| r8a.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r8a.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8a.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8a.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8a.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| r8a.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| r8a.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8a.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| r8a.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| r8a.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| r8a.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| r8g.sedang 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| r8g.besar 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8g.xbesar 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r8g.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8g.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8g.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| r8g.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| r8g.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8g.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| r8g.48xbesar 2 | 40000 | | 500.0 | | 240000 | |
| r8g.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| r8g.logam-48xl 2 | 40000 | | 500.0 | | 240000 | |
| r8gb.medium 1 | 1562 | 25000 | 195.31 | 3125.00 | 7500 | 120000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8gb.besar 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| r8gb.xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| r8gb.2xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |
| r8gb.4xbesar 2 | 25000 | | 3125.0 | | 120000 | |
| r8gb.8xbesar 2 | 5000 | | 6250.0 | | 240000 | |
| r8gb.12xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| r8gb.16xbesar 2 | 100000 | | 12500.0 | | 480000 | |
| r8gb.24xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| r8gb.48xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| r8gb.logam-24xl 2 | 150000 | | 18750.0 | | 72000 | |
| r8gb.logam-48xl 2 | 300000 | | 37500.0 | | 1440000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8gd.medium 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| r8gd.besar 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| r8gd.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r8gd.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8gd.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8gd.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| r8gd.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| r8gd.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8gd.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| r8gd.48xbesar 2 | 40000 | | 500.0 | | 240000 | |
| r8gd.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8gd.logam-48xl 2 | 40000 | | 500.0 | | 240000 | |
| r8gn.sedang 1 | 760 | 10000 | 95.00 | 1250.00 | 2500 | 40000 |
| r8gn.besar 1 | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| r8gn.xlarge 1 | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| r8gn.2xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8gn.4xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| r8gn.8xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8gn.12xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| r8gn.16xbesar 2 | 40000 | | 500.0 | | 160000 | |
| r8gn.24xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| r8gn.48xbesar 2 | 120000 | | 1500.0 | | 480000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8gn.logam-24xl 2 | 60000 | | 7500.0 | | 240000 | |
| r8gn.logam-48xl 2 | 120000 | | 1500.0 | | 480000 | |
| r8i.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r8i.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r8i.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8i.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8i.8xbesar 2 | 1000 | | 1250.0 | | 40000 | |
| r8i.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| r8i.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8i.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| r8i.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| r8i.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8i.96xbesar 2 | 80000 | | 1000.0 | | 480000 | |
| r8i.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| r8i.logam-96xl 2 | 80000 | | 1000.0 | | 480000 | |
| r8id.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r8id.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r8id.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8id.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8id.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| r8id.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| r8id.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8id.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8id.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| r8id.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| r8id.96xbesar 2 | 80000 | | 1000.0 | | 480000 | |
| r8id.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| r8id.logam-96xl 2 | 80000 | | 1000.0 | | 480000 | |
| r8i-flex.besar 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |
| r8i-flex.xlarge 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| r8i-flex.2xbesar 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r8i-flex.4xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8i-flex.8xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8i-flex.12xbesar 1 | 7500 | 15000 | 937.50 | 1875.00 | 30000 | 60000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8i-flex.16xbesar 1 | 10000 | 2000 | 1250.00 | 2500.00 | 40000 | 80000 |
| r8in.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r8in.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r8in.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8in.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8in.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| r8in.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| r8in.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8in.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| r8in.32xlarge 2 | 40000 | | 500.0 | | 160000 | |
| r8in.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8in.96xbesar 2 | 120000 | | 1500.0 | | 480000 | |
| r8in.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| r8in.logam-96xl 2 | 120000 | | 1500.0 | | 480000 | |
| r8idn.large 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| r8idn.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| r8idn.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| r8idn.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| r8idn.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| r8idn.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| r8idn.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| r8idn.24xlarge 2 | 30000 | | 3750.0 | | 120000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8idn.32xlarge 2 | 40000 | | 500.0 | | 160000 | |
| r8idn.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| r8idn.96xbesar 2 | 120000 | | 1500.0 | | 480000 | |
| r8idn.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| r8idn.logam-96xl 2 | 120000 | | 1500.0 | | 480000 | |
| r8ib.besar 1 | 1563 | 25000 | 195.38 | 3125.00 | 7500 | 120000 |
| r8ib.xbesar 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| r8ib.2xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| r8ib.4xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |
| r8ib.8xbesar 2 | 25000 | | 3125.0 | | 120000 | |
| r8ib.12xbesar 2 | 37500 | | 4687.5 | | 180000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8ib.16xbesar 2 | 5000 | | 6250.0 | | 240000 | |
| r8ib.24xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| r8ib.32xbesar 2 | 100000 | | 12500.0 | | 480000 | |
| r8ib.48xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| r8ib.96xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| r8ib.logam-48xl 2 | 150000 | | 18750.0 | | 72000 | |
| r8ib.logam-96xl 2 | 300000 | | 37500.0 | | 1440000 | |
| r8idb.besar 1 | 1563 | 25000 | 195.38 | 3125.00 | 7500 | 120000 |
| r8idb.xlarge 1 | 3125 | 25000 | 390.62 | 3125.00 | 15000 | 120000 |
| r8idb.2xbesar 1 | 6250 | 25000 | 781.25 | 3125.00 | 30000 | 120000 |
| r8idb.4xbesar 1 | 12500 | 25000 | 1562.50 | 3125.00 | 60000 | 120000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r8idb.8xbesar 2 | 25000 | | 3125.0 | | 120000 | |
| r8idb.12xbesar 2 | 37500 | | 4687.5 | | 180000 | |
| r8idb.16xbesar 2 | 5000 | | 6250.0 | | 240000 | |
| r8idb.24xbesar 2 | 75000 | | 9375.0 | | 360000 | |
| r8idb.32xbesar 2 | 100000 | | 12500.0 | | 480000 | |
| r8idb.48xbesar 2 | 150000 | | 18750.0 | | 72000 | |
| r8idb.96xbesar 2 | 300000 | | 37500.0 | | 1440000 | |
| r8idb.logam-48xl 2 | 150000 | | 18750.0 | | 72000 | |
| r8idb.logam-96xl 2 | 300000 | | 37500.0 | | 1440000 | |
| r9g.sedang 1 | 380 | 12000 | 47.50 | 150.00 | 2500 | 48000 |
| r9g.besar 1 | 760 | 12000 | 95.00 | 150.00 | 3600 | 48000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r9g.xlarge 1 | 1500 | 12000 | 187.50 | 150.00 | 6000 | 48000 |
| r9g.2xbesar 1 | 3000 | 12000 | 375.00 | 150.00 | 12000 | 48000 |
| r9g.4xbesar 1 | 6000 | 12000 | 750.00 | 150.00 | 24000 | 48000 |
| r9g.8xbesar 2 | 12000 | | 150.0 | | 48000 | |
| r9g.12xbesar 2 | 18000 | | 2250.0 | | 72000 | |
| r9g.16xbesar 2 | 24000 | | 3000.0 | | 96000 | |
| r9g.24xbesar 2 | 36000 | | 4500.0 | | 144000 | |
| r9g.48xbesar 2 | 72000 | | 900.0 | | 288000 | |
| r9g.logam-48xl 2 | 72000 | | 900.0 | | 288000 | |
| r9gd.medium 1 | 380 | 12000 | 47.50 | 150.00 | 2500 | 48000 |
| r9gd.besar 1 | 760 | 12000 | 95.00 | 150.00 | 3600 | 48000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| r9gd.xlarge 1 | 1500 | 12000 | 187.50 | 150.00 | 6000 | 48000 |
| r9gd.2xbesar 1 | 3000 | 12000 | 375.00 | 150.00 | 12000 | 48000 |
| r9gd.4xbesar 1 | 6000 | 12000 | 750.00 | 150.00 | 24000 | 48000 |
| r9gd.8xbesar 2 | 12000 | | 150.0 | | 48000 | |
| r9gd.12xbesar 2 | 18000 | | 2250.0 | | 72000 | |
| r9gd.16xbesar 2 | 24000 | | 3000.0 | | 96000 | |
| r9gd.24xbesar 2 | 36000 | | 4500.0 | | 144000 | |
| r9gd.48xbesar 2 | 72000 | | 900.0 | | 288000 | |
| r9gd.logam-48xl 2 | 72000 | | 900.0 | | 288000 | |
| u-3tb1.56xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| u-6tb1.56xlarge [2] | 38000 | | 4750.0 | | 160000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| u-6tb1.112xlarge [2] | 38000 | | 4750.0 | | 160000 | |
| u-6tb1.metal [2] | 38000 | | 4750.0 | | 160000 | |
| u-9tb1.112xlarge [2] | 38000 | | 4750.0 | | 160000 | |
| u-9tb1.metal [2] | 38000 | | 4750.0 | | 160000 | |
| u-12tb1.112xlarge [2] | 38000 | | 4750.0 | | 160000 | |
| u-12tb1.metal [2] | 38000 | | 4750.0 | | 160000 | |
| u-18tb1.112xlarge [2] | 38000 | | 4750.0 | | 160000 | |
| u-18tb1.metal [2] | 38000 | | 4750.0 | | 160000 | |
| u-24tb1.112xlarge [2] | 38000 | | 4750.0 | | 160000 | |
| u-24tb1.metal [2] | 38000 | | 4750.0 | | 160000 | |
| u7i-6tb.112xbesar 2 | 100000 | | 12500.0 | | 560000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| u7i-8tb.112xbesar 2 | 100000 | | 12500.0 | | 560000 | |
| u7i-12tb.224xbesar 2 | 100000 | | 12500.0 | | 560000 | |
| u7in-16tb.224xbesar 2 | 100000 | | 12500.0 | | 560000 | |
| u7in-24tb.224xbesar 2 | 100000 | | 12500.0 | | 560000 | |
| u7in-32tb.224xbesar 2 | 100000 | | 12500.0 | | 560000 | |
| u7inh-32tb.480xbesar 2 | 160000 | | 20000.0 | | 840000 | |
| x1.16xlarge [2] | 7000 | | 875.0 | | 40000 | |
| x1.32xlarge [2] | 14000 | | 1750.0 | | 80000 | |
| x1e.xlarge [2] | 500 | | 62.5 | | 3700 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| x1e.2xlarge [2] | 1000 | | 125.0 | | 7400 | |
| x1e.4xlarge [2] | 1750 | | 218.75 | | 10000 | |
| x1e.8xlarge [2] | 3500 | | 437.5 | | 2000 | |
| x1e.16xlarge [2] | 7000 | | 875.0 | | 40000 | |
| x1e.32xlarge [2] | 14000 | | 1750.0 | | 80000 | |
| x2gd.medium [1] | 315 | 4750 | 39.38 | 593.75 | 2500 | 2000 |
| x2gd.large [1] | 630 | 4750 | 78.75 | 593.75 | 3600 | 2000 |
| x2gd.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |
| x2gd.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| x2gd.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| x2gd.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| x2gd.12xlarge [2] | 14250 | | 1781.25 | | 60000 | |
| x2gd.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| x2gd.metal [2] | 19000 | | 2375.0 | | 80000 | |
| x2idn.16xlarge [2] | 40000 | | 500.0 | | 173333 | |
| x2idn.24xlarge [2] | 60000 | | 7500.0 | | 260000 | |
| x2idn.32xlarge [2] | 80000 | | 1000.0 | | 260000 | |
| x2idn.metal [2] | 80000 | | 1000.0 | | 260000 | |
| x2iedn.xlarge [1] | 2500 | 2000 | 312.50 | 2500.00 | 8125 | 65000 |
| x2iedn.2xlarge [1] | 5000 | 2000 | 625.00 | 2500.00 | 16250 | 65000 |
| x2iedn.4xlarge [1] | 10000 | 2000 | 1250.00 | 2500.00 | 32500 | 65000 |
| x2iedn.8xlarge [2] | 2000 | | 2500.0 | | 65000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| x2iedn.16xlarge [2] | 40000 | | 500.0 | | 130000 | |
| x2iedn.24xlarge [2] | 60000 | | 7500.0 | | 195000 | |
| x2iedn.32xlarge [2] | 80000 | | 1000.0 | | 260000 | |
| x2iedn.metal [2] | 80000 | | 1000.0 | | 260000 | |
| x2iezn.2xlarge [2] | 3170 | | 396.25 | | 13333 | |
| x2iezn.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| x2iezn.6xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| x2iezn.8xlarge [2] | 12000 | | 150.0 | | 55000 | |
| x2iezn.12xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| x2iezn.metal [2] | 19000 | | 2375.0 | | 80000 | |
| x8g.sedang 1 | 315 | 10000 | 39.38 | 1250.00 | 2500 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| x8g.besar 1 | 630 | 10000 | 78.75 | 1250.00 | 3600 | 40000 |
| x8g.xbesar 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| x8g.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |
| x8g.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| x8g.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| x8g.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| x8g.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| x8g.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| x8g.48xbesar 2 | 40000 | | 500.0 | | 240000 | |
| x8g.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| x8g.logam-48xl 2 | 40000 | | 500.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| x8aedz.large 1 | 1250 | 15000 | 156.25 | 1875.00 | 5000 | 60000 |
| x8aedz.xlarge 1 | 2500 | 15000 | 312.50 | 1875.00 | 10000 | 60000 |
| x8aedz.3xbesar 1 | 7500 | 15000 | 937.50 | 1875.00 | 30000 | 60000 |
| x8aedz.6xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| x8aedz.12xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| x8aedz.24xlarge 2 | 60000 | | 7500.0 | | 240000 | |
| x8aedz.logam-12xl 2 | 30000 | | 3750.0 | | 120000 | |
| x8aedz.logam-24xl 2 | 60000 | | 7500.0 | | 240000 | |
| x8i.besar 1 | 650 | 10000 | 81.25 | 1250.00 | 3600 | 40000 |
| x8i.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| x8i.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 12000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| x8i.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| x8i.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| x8i.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| x8i.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| x8i.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| x8i.32xbesar 2 | 40000 | | 500.0 | | 160000 | |
| x8i.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| x8i.64xbesar 2 | 70000 | | 8750.0 | | 32000 | |
| x8i.96xbesar 2 | 80000 | | 1000.0 | | 480000 | |
| x8i.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| x8i.logam-96xl 2 | 80000 | | 1000.0 | | 480000 | |
| z1d.large [1] | 800 | 3170 | 100.00 | 396.25 | 3333 | 13333 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| z1d.xlarge [1] | 1580 | 3170 | 197.50 | 396.25 | 6667 | 13333 |
| z1d.2xlarge [2] | 3170 | | 396.25 | | 13333 | |
| z1d.3xlarge [2] | 4750 | | 593.75 | | 2000 | |
| z1d.6xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| z1d.12xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| z1d.metal [2] | 19000 | | 2375.0 | | 80000 | |

[1] Instans ini dapat mempertahankan kinerja maksimum selama 30 menit setidaknya sekali setiap 24 jam, setelah itu mereka kembali ke kinerja dasar.

[2] Instans ini dapat mempertahankan performa yang dinyatakan tanpa batas waktu. Jika beban kerja Anda memerlukan performa maksimum yang berkelanjutan selama lebih dari 30 menit, gunakan salah satu instans ini.

Penyimpanan yang dioptimalkan

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| d2.xlarge [2] | 750 | | 93.75 | | 6000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| d2.2xlarge [2] | 1000 | | 125.0 | | 8000 | |
| d2.4xlarge [2] | 2000 | | 250.0 | | 16000 | |
| d2.8xlarge [2] | 4000 | | 500.0 | | 32000 | |
| d3.xlarge [1] | 850 | 2800 | 106.25 | 350.00 | 5000 | 15000 |
| d3.2xlarge [1] | 1700 | 2800 | 212.50 | 350.00 | 10000 | 15000 |
| d3.4xlarge [2] | 2800 | | 350.0 | | 15000 | |
| d3.8xlarge [2] | 5000 | | 625.0 | | 30000 | |
| d3en.xlarge [1] | 850 | 2800 | 106.25 | 350.00 | 5000 | 15000 |
| d3en.2xlarge [1] | 1700 | 2800 | 212.50 | 350.00 | 10000 | 15000 |
| d3en.4xlarge [2] | 2800 | | 350.0 | | 15000 | |
| d3en.6xlarge [2] | 4000 | | 500.0 | | 25000 | |
| d3en.8xlarge [2] | 5000 | | 625.0 | | 30000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| d3en.12xlarge [2] | 7000 | | 875.0 | | 40000 | |
| h1.2xlarge [2] | 1750 | | 218.75 | | 12000 | |
| h1.4xlarge [2] | 3500 | | 437.5 | | 2000 | |
| h1.8xlarge [2] | 7000 | | 875.0 | | 40000 | |
| h1.16xlarge [2] | 14000 | | 1750.0 | | 80000 | |
| i3.large [2] | 425 | | 53.125 | | 3000 | |
| i3.xlarge [2] | 850 | | 106.25 | | 6000 | |
| i3.2xlarge [2] | 1700 | | 212.5 | | 12000 | |
| i3.4xlarge [2] | 3500 | | 437.5 | | 16000 | |
| i3.8xlarge [2] | 7000 | | 875.0 | | 32500 | |
| i3.16xlarge [2] | 14000 | | 1750.0 | | 65000 | |
| i3.metal [2] | 19000 | | 2375.0 | | 80000 | |
| i3en.large [1] | 576 | 4750 | 72.10 | 593.75 | 3000 | 2000 |
| i3en.xlarge [1] | 1153 | 4750 | 144.20 | 593.75 | 6000 | 2000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| i3en.2xlarge [1] | 2307 | 4750 | 288.39 | 593.75 | 12000 | 2000 |
| i3en.3xlarge [1] | 3800 | 4750 | 475.00 | 593.75 | 15000 | 2000 |
| i3en.6xlarge [2] | 4750 | | 593.75 | | 2000 | |
| i3en.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| i3en.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| i3en.metal [2] | 19000 | | 2375.0 | | 80000 | |
| i4g.large [1] | 625 | 10000 | 78.12 | 1250.00 | 2500 | 40000 |
| i4g.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| i4g.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| i4g.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| i4g.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| i4g.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| i4i.large [1] | 625 | 10000 | 78.12 | 1250.00 | 2500 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| i4i.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| i4i.2xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| i4i.4xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| i4i.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| i4i.12xlarge [2] | 15000 | | 1875.0 | | 60000 | |
| i4i.16xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| i4i.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| i4i.32xlarge [2] | 40000 | | 500.0 | | 160000 | |
| i4i.metal [2] | 40000 | | 500.0 | | 160000 | |
| i7i.besar 1 | 625 | 10000 | 78.12 | 1250.00 | 2500 | 40000 |
| i7i.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| i7i.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| i7i.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| i7i.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| i7i.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| i7i.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| i7i.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| i7i.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| i7i.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| i7i.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| i7ie.large 1 | 625 | 10000 | 78.12 | 1250.00 | 2500 | 40000 |
| i7ie.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| i7ie.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| i7ie.3xbesar 1 | 3750 | 10000 | 468.75 | 1250.00 | 15000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| i7ie.6xbesar 1 | 7500 | 10000 | 937.50 | 1250.00 | 30000 | 40000 |
| i7ie.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| i7ie.18xbesar 2 | 22500 | | 2812.5 | | 90000 | |
| i7ie.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| i7ie.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| i7ie.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| i7ie.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| i8g.besar 1 | 625 | 10000 | 78.12 | 1250.00 | 2500 | 40000 |
| i8g.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| i8g.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| i8g.4xbesar 1 | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| i8g.8xbesar 2 | 10000 | | 1250.0 | | 40000 | |
| i8g.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| i8g.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| i8g.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| i8g.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| i8g.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| i8g.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| i8ge.large 1 | 625 | 10000 | 78.12 | 1250.00 | 2500 | 40000 |
| i8ge.xlarge 1 | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| i8ge.2xbesar 1 | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| i8ge.3xbesar 1 | 3750 | 10000 | 468.75 | 1250.00 | 15000 | 40000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| i8ge.6xbesar 1 | 7500 | 10000 | 937.50 | 1250.00 | 30000 | 40000 |
| i8ge.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| i8ge.18xbesar 2 | 22500 | | 2812.5 | | 90000 | |
| i8ge.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| i8ge.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| i8ge.logam-24xl 2 | 30000 | | 3750.0 | | 120000 | |
| i8ge.logam-48xl 2 | 60000 | | 7500.0 | | 240000 | |
| im4gn.large [1] | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| im4gn.xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| im4gn.2xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| im4gn.4xlarge [2] | 10000 | | 1250.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| im4gn.8xlarge [2] | 2000 | | 2500.0 | | 80000 | |
| im4gn.16xlarge [2] | 40000 | | 500.0 | | 160000 | |
| is4gen.medium [1] | 625 | 10000 | 78.12 | 1250.00 | 2500 | 40000 |
| is4gen.large [1] | 1250 | 10000 | 156.25 | 1250.00 | 5000 | 40000 |
| is4gen.xlarge [1] | 2500 | 10000 | 312.50 | 1250.00 | 10000 | 40000 |
| is4gen.2xlarge [1] | 5000 | 10000 | 625.00 | 1250.00 | 2000 | 40000 |
| is4gen.4xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| is4gen.8xlarge [2] | 2000 | | 2500.0 | | 80000 | |

[1] Instans ini dapat mempertahankan kinerja maksimum selama 30 menit setidaknya sekali setiap 24 jam, setelah itu mereka kembali ke kinerja dasar.

[2] Instans ini dapat mempertahankan performa yang dinyatakan tanpa batas waktu. Jika beban kerja Anda memerlukan performa maksimum yang berkelanjutan selama lebih dari 30 menit, gunakan salah satu instans ini.

Komputasi yang dipercepat

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| dl1.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| dl2q.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| f2.6xbesar 2 | 7500 | | 937.5 | | 30000 | |
| f2.12xbesar 2 | 15000 | | 1875.0 | | 60000 | |
| f2.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| g3.4xlarge [2] | 3500 | | 437.5 | | 2000 | |
| g3.8xlarge [2] | 7000 | | 875.0 | | 40000 | |
| g3.16xlarge [2] | 14000 | | 1750.0 | | 80000 | |
| g4ad.xlarge [1] | 400 | 3170 | 50.00 | 396.25 | 1700 | 13333 |
| g4ad.2xlarge [1] | 800 | 3170 | 100.00 | 396.25 | 3400 | 13333 |
| g4ad.4xlarge [1] | 1580 | 3170 | 197.50 | 396.25 | 6700 | 13333 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| g4ad.8xlarge [2] | 3170 | | 396.25 | | 13333 | |
| g4ad.16xlarge [2] | 6300 | | 787.5 | | 26667 | |
| g4dn.xlarge [1] | 950 | 3500 | 118.75 | 437.50 | 3000 | 2000 |
| g4dn.2xlarge [1] | 1150 | 3500 | 143.75 | 437.50 | 6000 | 2000 |
| g4dn.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| g4dn.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| g4dn.12xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| g4dn.16xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| g4dn.metal [2] | 19000 | | 2375.0 | | 80000 | |
| g5.xlarge [1] | 700 | 3500 | 87.50 | 437.50 | 3000 | 15000 |
| g5.2xlarge [1] | 850 | 3500 | 106.25 | 437.50 | 3500 | 15000 |
| g5.4xlarge [2] | 4750 | | 593.75 | | 2000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| g5.8xlarge [2] | 16000 | | 2000.0 | | 65000 | |
| g5.12xlarge [2] | 16000 | | 2000.0 | | 65000 | |
| g5.16xlarge [2] | 16000 | | 2000.0 | | 65000 | |
| g5.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| g5.48xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| g5g.xlarge [1] | 1188 | 4750 | 148.50 | 593.75 | 6000 | 2000 |
| g5g.2xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 12000 | 2000 |
| g5g.4xlarge [2] | 4750 | | 593.75 | | 2000 | |
| g5g.8xlarge [2] | 9500 | | 1187.5 | | 40000 | |
| g5g.16xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| g5g.metal [2] | 19000 | | 2375.0 | | 80000 | |
| g6.xbesar 1 | 1000 | 5000 | 125.00 | 625.00 | 4000 | 2000 |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| g6.2xbesar 1 | 2000 | 5000 | 250.00 | 625.00 | 8000 | 2000 |
| g6.4xbesar 2 | 8000 | | 1000.0 | | 32000 | |
| g6.8xbesar 2 | 16000 | | 2000.0 | | 64000 | |
| g6.12xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| g6.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| g6.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| g6.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| g6e.xlarge 1 | 1000 | 5000 | 125.00 | 625.00 | 4000 | 2000 |
| g6e.2xbesar 1 | 2000 | 5000 | 250.00 | 625.00 | 8000 | 2000 |
| g6e.4xbesar 2 | 8000 | | 1000.0 | | 32000 | |
| g6e.8xbesar 2 | 16000 | | 2000.0 | | 64000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| g6e.12xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| g6e.16xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| g6e.24xbesar 2 | 30000 | | 3750.0 | | 120000 | |
| g6e.48xbesar 2 | 60000 | | 7500.0 | | 240000 | |
| g6f.besar 1 | 936 | 5000 | 117.00 | 625.00 | 3750 | 2000 |
| g6f.xlarge 1 | 1000 | 5000 | 125.00 | 625.00 | 4000 | 2000 |
| g6f.2xbesar 1 | 2000 | 5000 | 250.00 | 625.00 | 8000 | 2000 |
| g6f.4xbesar 2 | 6000 | | 750.0 | | 24000 | |
| gr6.4xbesar 2 | 8000 | | 1000.0 | | 32000 | |
| gr6.8xbesar 2 | 16000 | | 2000.0 | | 64000 | |
| gr6f.4xbesar 2 | 8000 | | 1000.0 | | 32000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| g7.2xbesar 1 | 4000 | 8000 | 500.00 | 1000.00 | 16000 | 32000 |
| g7.4xbesar 2 | 8000 | | 1000.0 | | 32000 | |
| g7.8xbesar 2 | 16000 | | 2000.0 | | 64000 | |
| g7.12xbesar 2 | 2000 | | 2500.0 | | 80000 | |
| g7.24xbesar 2 | 40000 | | 500.0 | | 160000 | |
| g7.48xbesar 2 | 80000 | | 1000.0 | | 32000 | |
| g7e.2xbesar 1 | 2000 | 5000 | 250.00 | 625.00 | 8000 | 2000 |
| g7e.4xbesar 2 | 8000 | | 1000.0 | | 32000 | |
| g7e.8xbesar 2 | 16000 | | 2000.0 | | 64000 | |
| g7e.12xbesar 2 | 25000 | | 3125.0 | | 100000 | |
| g7e.24xbesar 2 | 5000 | | 6250.0 | | 200000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| g7e.48xlarge 2 | 100000 | | 12500.0 | | 400000 | |
| inf1.xlarge [1] | 1190 | 4750 | 148.75 | 593.75 | 4000 | 2000 |
| inf1.2xlarge [1] | 1190 | 4750 | 148.75 | 593.75 | 6000 | 2000 |
| inf1.6xlarge [2] | 4750 | | 593.75 | | 2000 | |
| inf1.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| inf2.xlarge [1] | 1250 | 10000 | 156.25 | 1250.00 | 6000 | 40000 |
| inf2.8xlarge [2] | 10000 | | 1250.0 | | 40000 | |
| inf2.24xlarge [2] | 30000 | | 3750.0 | | 120000 | |
| inf2.48xlarge [2] | 60000 | | 7500.0 | | 240000 | |
| p3.2xlarge [2] | 1750 | | 218.75 | | 10000 | |
| p3.8xlarge [2] | 7000 | | 875.0 | | 40000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| p3.16xlarge [2] | 14000 | | 1750.0 | | 80000 | |
| p3dn.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| p4d.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| p4de.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |
| p5.4xbesar 2 | 10000 | | 1250.0 | | 32500 | |
| p5.48xlarge [2] | 80000 | | 1000.0 | | 260000 | |
| p5e.48xlarge 2 | 80000 | | 1000.0 | | 260000 | |
| p5en.48xbesar 2 | 100000 | | 12500.0 | | 400000 | |
| p6-b200.48xbesar 2 | 100000 | | 12500.0 | | 400000 | |
| p6-b300.48xbesar 2 | 100000 | | 12500.0 | | 400000 | |
| p6e-gb200.36xbesar 2 | 60000 | | 7500.0 | | 240000 | |

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| trn1.2xlarge [1] | 5000 | 2000 | 625.00 | 2500.00 | 16250 | 65000 |
| trn1.32xlarge [2] | 80000 | | 1000.0 | | 260000 | |
| trn1n.32xlarge [2] | 80000 | | 1000.0 | | 260000 | |
| trn2.3xbesar 2 | 5000 | | 625.0 | | 16250 | |
| trn2.48xbesar 2 | 80000 | | 1000.0 | | 260000 | |
| trn2u.48xlarge 2 | 80000 | | 1000.0 | | 260000 | |
| vt1.3xlarge [1] | 2375 | 4750 | 296.88 | 593.75 | 10000 | 2000 |
| vt1.6xlarge [2] | 4750 | | 593.75 | | 2000 | |
| vt1.24xlarge [2] | 19000 | | 2375.0 | | 80000 | |

[1] Instans ini dapat mempertahankan kinerja maksimum selama 30 menit setidaknya sekali setiap 24 jam, setelah itu mereka kembali ke kinerja dasar.

[2] Instans ini dapat mempertahankan performa yang dinyatakan tanpa batas waktu. Jika beban kerja Anda memerlukan performa maksimum yang berkelanjutan selama lebih dari 30 menit, gunakan salah satu instans ini.

High-performance komputasi

| Ukuran instans | Bandwidth dasar (Mbps) | Bandwidth maksimum (Mbps) | Throughput dasar (MB/s, 128 KiB I/O) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS dasar (16 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|---|---|---|
| hpc6a.48xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc6id.32xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc7a.12xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc7a.24xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc7a.48xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc7a.96xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc7g.4xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc7g.8xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc7g.16xlarge [1] | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |
| hpc8a.96xbesar 1 | 87 | 2085 | 10.88 | 260.62 | 500 | 11000 |

[1] Instans ini dapat mempertahankan kinerja maksimum selama 30 menit setidaknya sekali setiap 24 jam, setelah itu mereka kembali ke kinerja dasar.

[2] Instans ini dapat mempertahankan performa yang dinyatakan tanpa batas waktu. Jika beban kerja Anda memerlukan performa maksimum yang berkelanjutan selama lebih dari 30 menit, gunakan salah satu instans ini.

## Optimisasi EBS didukung

Jenis instans berikut mendukung pengoptimalan EBS tetapi pengoptimalan EBS tidak diaktifkan secara default. Anda harus mengaktifkan pengoptimalan EBS, dengan biaya tambahan per jam, selama atau setelah peluncuran untuk mencapai tingkat kinerja EBS yang dijelaskan.

| Ukuran instans | Bandwidth maksimum (Mbps) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|
| c1.xlarge | 1000 | 125.0 | 8000 |
| c3.xlarge | 500 | 62.5 | 4000 |
| c3.2xlarge | 1000 | 125.0 | 8000 |
| c3.4xlarge | 2000 | 250.0 | 16000 |
| i2.xlarge | 500 | 62.5 | 4000 |
| i2.2xlarge | 1000 | 125.0 | 8000 |
| i2.4xlarge | 2000 | 250.0 | 16000 |
| m1.large | 500 | 62.5 | 4000 |
| m1.xlarge | 1000 | 125.0 | 8000 |
| m2.2xlarge | 500 | 62.5 | 4000 |
| m2.4xlarge | 1000 | 125.0 | 8000 |
| m3.xlarge | 500 | 62.5 | 4000 |
| m3.2xlarge | 1000 | 125.0 | 8000 |
| r3.xlarge | 500 | 62.5 | 4000 |

| Ukuran instans | Bandwidth maksimum (Mbps) | Throughput maksimum (MB/s, 128 KiB I/O) | IOPS maksimum (16 KiB I/O) |
|---|---|---|---|
| r3.2xlarge | 1000 | 125.0 | 8000 |
| r3.4xlarge | 2000 | 250.0 | 16000 |

**Note**

Instans `i2.8xlarge`, `c3.8xlarge`, dan `r3.8xlarge` tidak memiliki bandwidth EBS khusus dan tidak menawarkan optimisasi EBS. Pada instans ini, lalu lintas jaringan dan lalu lintas Amazon EBS berbagi antarmuka jaringan 10 gigabit yang sama.

## Dapatkan kinerja maksimum yang dioptimalkan Amazon EBS

Kinerja EBS instans dibatasi oleh batas kinerja jenis instans, atau kinerja agregat dari volume terlampirnya, mana saja yang lebih kecil. Untuk mencapai performa EBS maksimum, instans harus memiliki volume terlampir yang memberikan performa gabungan yang sama atau lebih besar dari performa instans maksimum. Misalnya, untuk mencapai `80,000` IOPS untuk`r6i.16xlarge`, instance harus memiliki setidaknya 5 `gp2` volume yang disediakan dengan masing-masing IOPS (5volume x `16,000` IOPS = `16,000` `80,000` IOPS), atau dapat memiliki 1 `gp3` volume yang disediakan dengan IOPS. `80,000` Sebaiknya pilih jenis instans yang menyediakan throughput Amazon EBS yang lebih khusus daripada yang dibutuhkan aplikasi Anda; jika tidak, koneksi antara Amazon EBS dan Amazon EC2 dapat menjadi hambatan kinerja.

**Important**

Saat menggunakan bobot bandwidth yang dapat dikonfigurasi, batas bandwidth EBS untuk instans Anda mungkin berubah. Untuk contoh dengan konfigurasi `VPC-1` bobot, yang meningkatkan bandwidth jaringan, Anda mungkin mengalami IOPS yang lebih rendah dari yang diharapkan untuk volume EBS karena mencapai batas bandwidth EBS sebelum batas IOPS. Ini terutama terlihat dengan I/O ukuran yang lebih besar. Selalu uji beban kerja spesifik Anda untuk memastikannya memenuhi persyaratan kinerja Anda dengan pembobotan

bandwidth yang Anda pilih. Untuk informasi selengkapnya, lihat Konfigurasi pembobotan bandwidth instans EC2.

Anda dapat menggunakan metrik `EBSIOBalance%` dan `EBSByteBalance%` untuk membantu Anda menentukan apakah instans Anda memiliki ukuran yang tepat. Anda dapat melihat metrik ini di CloudWatch konsol dan mengatur alarm yang dipicu berdasarkan ambang batas yang Anda tentukan. Metrik ini dinyatakan sebagai persentase. Instans dengan persentase keseimbangan yang rendah secara konsisten adalah kandidat yang harus naik ukurannya. Instans yang persentase keseimbangan tidak pernah turun di bawah 100% adalah kandidat untuk penurunan ukuran. Untuk informasi selengkapnya, lihat Memantau instans Anda menggunakan CloudWatch.

Instans memori yang tinggi dirancang untuk menjalankan basis data dalam memori yang besar, termasuk deployment produksi dari basis data dalam memori SAP HANA, di cloud. Untuk memaksimalkan performa EBS, gunakan instans memori yang tinggi dengan menerapkan jumlah genap volume `io1` atau `io2` dengan performa identik yang disediakan. Misalnya, untuk beban kerja berat IOPS, gunakan empat volume `io1` atau `io2` dengan 40.000 IOPS yang Tersedia untuk mendapatkan maksimum 160.000 instans IOPS. Demikian pula, untuk beban kerja throughput yang berat, gunakan enam `io1` atau `io2` volume dengan 48.000 IOPS yang disediakan untuk mendapatkan throughput maksimum 4.750. MB/s Untuk rekomendasi tambahan, lihat Konfigurasi Penyimpanan untuk SAP HANA.

Pertimbangan-pertimbangan

- Instans G4dn, i3en, Inf1, M5a, M5ad, R5a, R5ad, T3, T3a, dan Z1d yang diluncurkan setelah 26 Februari 2020 memberikan kinerja EBS maksimum yang dioptimalkan. Untuk mendapatkan performa maksimum dari suatu instans yang diluncurkan sebelum 26 Februari 2020, hentikan dan mulai.
- Instans C5, C5d, C5n, M5, M5d, M5n, M5dn, R5, R5d, R5n, R5dn, dan P3dn yang diluncurkan setelah 3 Desember 2019 memberikan kinerja maksimum yang dioptimalkan EBS. Untuk mendapatkan performa maksimum dari instans yang diluncurkan sebelum 3 Desember 2019, hentikan dan mulai.
- `u-6tb1.metal`,`u-9tb1.metal`, dan `u-12tb1.metal` instans yang diluncurkan setelah 12 Maret 2020 memberikan kinerja EBS maksimum yang dioptimalkan. Tipe instans ini diluncurkan sebelum 12 Maret 2020 mungkin memberikan performa yang lebih rendah. Untuk mendapatkan performa maksimum dari suatu instans yang diluncurkan sebelum 12 Maret 2020, hubungi tim akun Anda untuk mempebarui instansnya tanpa biaya tambahan.

## Temukan jenis instans EBS-optimized EC2

Anda dapat melihat jenis instans yang mendukung pengoptimalan EBS di setiap Wilayah.

### Console

Untuk menemukan jenis instans yang EBS-optimized secara default

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Tipe Instans.
3. Tambahkan filter EBS Optimization support = default.
4. (Opsional) Pilih ikon Preferensi dan kemudian nyalakan kolom yang relevan, seperti EBS Maximum IOPS dan EBS Baseline IOPS.
5. (Opsional) Tambahkan filter ke cakupan lebih lanjut ke jenis instans tertentu yang menarik.

### AWS CLI

Untuk menemukan jenis instans yang EBS-optimized secara default

Gunakan perintah describe-instance-types berikut ini.

```
aws ec2 describe-instance-types \
--filters Name=ebs-info.ebs-optimized-support,Values=default  \
--query 'InstanceTypes[].{InstanceType:InstanceType,
 "MaxBandwidth(Mb/s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,
 MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIops, "MaxThroughput(MB/
s)":EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--output=table
```

Untuk menemukan jenis instans yang secara opsional mendukung pengoptimalan EBS

Gunakan perintah describe-instance-types berikut ini.

```
aws ec2 describe-instance-types \
--filters Name=ebs-info.ebs-optimized-support,Values=supported \
--query 'InstanceTypes[].{InstanceType:InstanceType,
 "MaxBandwidth(Mb/s)":EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps,
 MaxIOPS:EbsInfo.EbsOptimizedInfo.MaximumIops, "MaxThroughput(MB/
s)":EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps}' \
--output=table
```

Berikut ini adalah contoh output untuk`eu-west-1`.

```
-------------------------------------------------------------------------
|                        DescribeInstanceTypes                          |
+--------------+----------------------+----------+----------------------+
| InstanceType | MaxBandwidth(Mb/s)   | MaxIOPS  |  MaxThroughput(MB/s) |
+--------------+----------------------+----------+----------------------+
|  i2.2xlarge  |  1000                |  8000    |  125.0               |
|  m2.4xlarge  |  1000                |  8000    |  125.0               |
|  m2.2xlarge  |  500                 |  4000    |  62.5                |
|  c1.xlarge   |  1000                |  8000    |  125.0               |
|  i2.xlarge   |  500                 |  4000    |  62.5                |
|  m3.xlarge   |  500                 |  4000    |  62.5                |
|  m1.xlarge   |  1000                |  8000    |  125.0               |
|  r3.4xlarge  |  2000                |  16000   |  250.0               |
|  r3.2xlarge  |  1000                |  8000    |  125.0               |
|  c3.xlarge   |  500                 |  4000    |  62.5                |
|  m3.2xlarge  |  1000                |  8000    |  125.0               |
|  r3.xlarge   |  500                 |  4000    |  62.5                |
|  i2.4xlarge  |  2000                |  16000   |  250.0               |
|  c3.4xlarge  |  2000                |  16000   |  250.0               |
|  c3.2xlarge  |  1000                |  8000    |  125.0               |
|  m1.large    |  500                 |  4000    |  62.5                |
+--------------+----------------------+----------+----------------------+
```

PowerShell

Untuk menemukan jenis instans yang EBS-optimized secara default

Gunakan Get-EC2InstanceType cmdlet.

```
Get-EC2InstanceType `
    -Filter @{Name="ebs-info.ebs-optimized-support"; Values="default"} | `
    Select InstanceType, `
        @{Name="MaxBandwidth(Mb/s)";
 Expression={($_.EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps)}}, `
        @{Name="MaxIOPS"; Expression={($_.EbsInfo.EbsOptimizedInfo.MaximumIops)}}, `
        @{Name="MaxThroughput (MB/s)";
 Expression={($_.EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps)}}
```

Untuk menemukan jenis instans yang secara opsional mendukung pengoptimalan EBS

Gunakan Get-EC2InstanceType cmdlet.

```
Get-EC2InstanceType `
    -Filter @{Name="ebs-info.ebs-optimized-support"; Values="supported"} | `
    Select InstanceType, `
        @{Name="MaxBandwidth(Mb/s)";
 Expression={($_.EbsInfo.EbsOptimizedInfo.MaximumBandwidthInMbps)}}, `
        @{Name="MaxIOPS"; Expression={($_.EbsInfo.EbsOptimizedInfo.MaximumIops)}}, `
        @{Name="MaxThroughput (MB/s)";
 Expression={($_.EbsInfo.EbsOptimizedInfo.MaximumThroughputInMBps)}}
```

Berikut ini adalah contoh output untuk`eu-west-1`.

```
InstanceType MaxBandwidth(Mb/s) MaxIOPS MaxThroughput (MB/s)
------------ ------------------ ------- --------------------
m2.4xlarge                 1000    8000              125.000
i2.2xlarge                 1000    8000              125.000
c1.xlarge                  1000    8000              125.000
m2.2xlarge                  500    4000               62.500
r3.2xlarge                 1000    8000              125.000
m3.xlarge                   500    4000               62.500
r3.4xlarge                 2000   16000              250.000
m1.xlarge                  1000    8000              125.000
i2.xlarge                   500    4000               62.500
c3.xlarge                   500    4000               62.500
c3.4xlarge                 2000   16000              250.000
c3.2xlarge                 1000    8000              125.000
i2.4xlarge                 2000   16000              250.000
r3.xlarge                   500    4000               62.500
m3.2xlarge                 1000    8000              125.000
m1.large                    500    4000               62.500
```

## Aktifkan pengoptimalan EBS untuk instans Amazon EC2

Anda dapat mengaktifkan pengoptimalan EBS secara manual hanya untuk jenis instans generasi sebelumnya yang secara opsional mendukung pengoptimalan EBS. Jika Anda mengaktifkan pengoptimalan EBS untuk jenis instans ini, ada biaya tambahan per jam

Prasyarat

- Pastikan jenis instans mengharuskan Anda mengaktifkan pengoptimalan EBS. Untuk informasi selengkapnya, lihat Optimisasi EBS didukung.
- Untuk mengaktifkan pengoptimalan EBS setelah peluncuran, Anda harus menghentikan instans.

> **Warning**
>
> Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

### Console

Untuk mengaktifkan pengoptimalan Amazon EBS selama peluncuran

Di wizard Luncurkan instance, pilih jenis instans yang diperlukan. Perluas bagian Detail lanjutan, lalu EBS-optimized misalnya, pilih Aktifkan.

Jika jenis instans yang dipilih tidak mendukung pengoptimalan Amazon EBS, drop-down dinonaktifkan. Jika tipe instansnya adalah Amazon EBS-optimized secara default, Aktifkan sudah dipilih.

Untuk mengaktifkan pengoptimalan Amazon EBS setelah peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, dan pilih instans.
3. Hentikan instans. Pilih T ind akan, Status Inst ance, Hentikan instance.
4. Dengan instans yang masih dipilih, klik Tindakan, Pengaturan instans, Ubah tipe instans.
5. Pilih EBS-optimized dan kemudian pilih Ter apkan.

   Jika tipe instans adalah Amazon secara EBS-optimized default, atau jika tidak mendukung pengoptimalan Amazon EBS, kotak centang dinonaktifkan.
6. Mulai ulang instans. Pilih Status instans, Mulai instans.

### AWS CLI

Untuk mengaktifkan pengoptimalan Amazon EBS selama peluncuran

Gunakan perintah run-instance dengan opsi. `--ebs-optimized`

Untuk mengaktifkan pengoptimalan Amazon EBS setelah peluncuran

1. Jika instance sedang berjalan, hentikan dengan menggunakan perintah stop-instance.

```
aws ec2 stop-instances --instance-ids i-1234567890abcdef0
```

2. Aktifkan optimasi EBS dengan menggunakan perintah modify-instance- atribut dengan opsi. `--ebs-optimized`

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --ebs-optimized
```

PowerShell

Untuk mengaktifkan pengoptimalan Amazon EBS selama peluncuran

Gunakan New-EC2Instance cmdlet dengan opsi. `-EbsOptimized`

Untuk mengaktifkan pengoptimalan Amazon EBS setelah peluncuran

1. Jika instance sedang berjalan, hentikan dengan menggunakan Stop-EC2Instance cmdlet.

```
Stop-EC2Instance -InstanceId i-1234567890abcdef0
```

2. Aktifkan optimasi EBS dengan menggunakan Edit-EC2InstanceAttribute cmdlet dengan opsi. `-EbsOptimized`

```
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -EbsOptimized $true
```

## Opsi CPU untuk instans Amazon EC2

Banyak instans Amazon EC2 mendukung multithreading simultan (SMT), yang memungkinkan beberapa utas berjalan secara bersamaan pada satu inti CPU. Setiap thread direpresentasikan sebagai CPU virtual (vCPU) pada instans. Sebuah instans memiliki jumlah inti CPU default, yang bervariasi sesuai dengan tipe instans. Misalnya, tipe instans `m5.xlarge` memiliki dua inti CPU dan dua thread per inti secara default—empat vCPU secara keseluruhan.

Dalam kebanyakan kasus, ada tipe instans Amazon EC2 yang memiliki kombinasi memori dan jumlah vCPU agar sesuai dengan beban kerja Anda. Namun, untuk mengoptimalkan instance untuk

beban kerja atau kebutuhan bisnis tertentu, Anda dapat menentukan opsi CPU berikut selama dan setelah peluncuran instance:

- Jumlah inti CPU: Anda dapat menyesuaikan jumlah inti CPU untuk instans. Anda mungkin akan melakukan ini agar dapat mengoptimalkan biaya lisensi perangkat lunak Anda dengan instans yang memiliki jumlah RAM yang cukup untuk beban kerja yang membutuhkan memori intensif tetapi dengan inti CPU yang lebih sedikit.
- Thread per core: Anda dapat menonaktifkan SMT dengan menentukan satu thread per inti CPU. Anda dapat melakukannya untuk beban kerja tertentu, seperti beban kerja komputasi performa tinggi (HPC).

Pertimbangan-pertimbangan

- Anda tidak dapat mengubah jumlah utas per inti untuk instans Mac silikon T2, C7a, M7a, R7a, dan Apple, dan instans berdasarkan prosesor Graviton. AWS
- Jumlah instance yang dapat Anda jalankan didasarkan pada vCPU default untuk jenis instans yang digunakan. Cara kami menghitung vCPU yang dikonsumsi oleh instance tidak terpengaruh dengan mengubah opsi CPU-nya.

Harga

Tidak ada biaya tambahan untuk menentukan opsi CPU. Untuk instans EC2 yang diluncurkan dari AMI Windows dan SQL Server yang disertakan lisensi, Anda dapat menyesuaikan opsi CPU untuk memanfaatkan fitur EC2 Optimize CPU untuk membayar biaya lisensi berdasarkan jumlah vCPU yang dikonfigurasi untuk instance Anda. Untuk instans EC2 lainnya, Anda dikenakan biaya yang sama dengan instans yang diluncurkan dengan opsi CPU default.

Daftar Isi


- Aturan untuk menentukan opsi CPU untuk instans Amazon EC2
- Opsi CPU yang didukung untuk jenis instans Amazon EC2
- Tentukan opsi CPU untuk instans Amazon EC2
- Melihat utas dan inti CPU untuk instans Amazon EC2
- Optimalkan CPU untuk instans License-Included

## Aturan untuk menentukan opsi CPU untuk instans Amazon EC2

Untuk menentukan opsi CPU untuk instans Anda, perhatikan aturan berikut:

- Anda tidak dapat menentukan opsi CPU untuk instans bare metal.
- Anda dapat menentukan opsi CPU baik selama dan setelah peluncuran instance.
- Untuk mengkonfigurasi opsi CPU, Anda harus menentukan jumlah inti CPU dan utas per inti dalam permintaan. Untuk contoh permintaan, lihat Tentukan opsi CPU untuk instans Amazon EC2.
- Jumlah vCPU untuk instans adalah jumlah inti CPU dikalikan dengan thread per inti. Untuk menentukan jumlah kustom vCPU, Anda harus menentukan jumlah inti CPU dan thread yang valid per inti untuk tipe instans tersebut. Anda tidak boleh melebihi jumlah default vCPU untuk instans. Untuk informasi selengkapnya, lihat Opsi CPU yang didukung untuk jenis instans Amazon EC2.
- Untuk menonaktifkan multithreading simultan (SMT), juga disebut sebagai hyper-threading, tentukan satu thread per inti.
- Di konsol, saat Anda mengubah jenis instans inst ans yang ada, Amazon EC2 menerapkan pengaturan opsi CPU dari instans yang ada ke instans baru, jika memungkinkan. Jika jenis instance baru tidak mendukung pengaturan tersebut, opsi CPU diatur ulang ke Gunakan opsi CPU default. Opsi ini menggunakan jumlah default vCPU untuk jenis instans baru.

  Untuk memperbarui pengaturan untuk instance baru, pilih T entukan opsi CPU di bawah Detail lanjutan di tampilan Ubah jenis instans.
- Opsi CPU yang ditentukan tidak akan berubah setelah Anda menghentikan, memulai, atau me-reboot sebuah instans.
- Jika Anda menggunakan Instans Cadangan, diskon mungkin tidak diterapkan saat Anda mengonfigurasi Optimalkan CPU untuk instans yang diluncurkan dari AMI Windows yang disertakan lisensi di akun pembayar yang sama. Kami menyarankan Anda menggunakan Paket Tabungan untuk mengurangi biaya CPU-based lisensi v dan memberikan penghematan yang sebanding pada biaya komputasi Anda.
- Untuk menghemat biaya lisensi untuk instans yang diluncurkan dari AMI yang disertakan lisensi Windows dan SQL Server, Anda harus mengonfigurasi minimal empat vCPU. Jika Anda mengonfigurasi kurang dari empat vCPU, penagihan default diterapkan.
- Mengoptimalkan CPU untuk License-Included instans tidak didukung pada tipe instans T3.
- Mengoptimalkan CPU untuk License-Included instans tidak didukung pada Instans Khusus.

# Opsi CPU yang didukung untuk jenis instans Amazon EC2

Tabel berikut mencantumkan tipe instans yang mendukung penentuan opsi CPU.

Daftar Isi


- Instans tujuan umum
- Instans komputasi yang dioptimalkan
- Instans memori yang dioptimalkan
- Instans penyimpanan yang dioptimalkan
- Instans komputasi terakselerasi
- High-performance instans komputasi


## Instans tujuan umum

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m2.xlarge | 2 | 2 | 1 | 1, 2 | 1 |
| m2.2xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m2.4xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m3.large | 2 | 1 | 2 | 1 | 1, 2 |
| m3.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m3.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m4.large | 2 | 1 | 2 | 1 | 1, 2 |
| m4.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m4.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m4.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m4.10xlarge | 40 | 20 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20 | 1, 2 |
| m4.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m5.large | 2 | 1 | 2 | 1 | 1, 2 |
| m5.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| m5.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m5.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m5.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m5.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m5.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m5.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| m5a.large | 2 | 1 | 2 | 1 | 1, 2 |
| m5a.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| m5a.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m5a.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m5a.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m5a.12xlarge | 48 | 24 | 2 | 6, 12, 18, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m5a.16xlarge | 64 | 32 | 2 | 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m5a.24xlarge | 96 | 48 | 2 | 12, 18, 24, 36, 48 | 1, 2 |
| m5ad.large | 2 | 1 | 2 | 1 | 1, 2 |
| m5ad.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| m5ad.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m5ad.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m5ad.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m5ad.12xlarge | 48 | 24 | 2 | 6, 12, 18, 24 | 1, 2 |
| m5ad.16xlarge | 64 | 32 | 2 | 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m5ad.24xlarge | 96 | 48 | 2 | 12, 18, 24, 36, 48 | 1, 2 |
| m5d.large | 2 | 1 | 2 | 1 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m5d.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| m5d.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m5d.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m5d.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m5d.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| m5d.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m5d.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| m5dn.large | 2 | 1 | 2 | 1 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m5dn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m5dn.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m5dn.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m5dn.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m5dn.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| m5dn.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m5dn.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| m5n.large | 2 | 1 | 2 | 1 | 1, 2 |
| m5n.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m5n.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m5n.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m5n.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m5n.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| m5n.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m5n.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| m5zn.large | 2 | 1 | 2 | 1 | 1, 2 |
| m5zn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m5zn.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m5zn.3xlarge | 12 | 6 | 2 | 2, 4, 6 | 1, 2 |
| m5zn.6xlarge | 24 | 12 | 2 | 2, 4, 6, 8, 10, 12 | 1, 2 |
| m5zn.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| m6a.large | 2 | 1 | 2 | 1 | 1, 2 |
| m6a.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m6a.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m6a.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m6a.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m6a.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24 | 1, 2 |
| m6a.16xlarge | 64 | 32 | 2 | 4, 6, 8, 10, 12, 14, 16, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6a.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 32, 48 | 1, 2 |
| m6a.32xlarge | 128 | 64 | 2 | 8, 12, 16, 20, 24, 28, 32, 64 | 1, 2 |
| m6a.48xlarge | 192 | 96 | 2 | 8, 12, 16, 20, 24, 28, 32, 64, 96 | 1, 2 |
| m6g.large | 2 | 2 | 1 | 1, 2 | 1 |
| m6g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m6g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m6g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m6g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| m6gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| m6gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m6gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m6gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| m6gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| m6i.large | 2 | 1 | 2 | 1 | 1, 2 |
| m6i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m6i.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m6i.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m6i.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6i.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| m6i.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m6i.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| m6i.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m6id.large | 2 | 1 | 2 | 1 | 1, 2 |
| m6id.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6id.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| m6id.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| m6id.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| m6id.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| m6id.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| m6id.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6id.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m6idn.large | 2 | 1 | 2 | 1 | 1, 2 |
| m6idn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m6idn.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m6idn.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m6idn.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6idn.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m6idn.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m6idn.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6idn.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m6in.large | 2 | 1 | 2 | 1 | 1, 2 |
| m6in.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m6in.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m6in.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m6in.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6in.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m6in.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m6in.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m6in.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m7a.large | 2 | 2 | 1 | 1, 2 | 1 |
| m7a.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m7a.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m7a.4xlarge | 16 | 16 | 1 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1 |
| m7a.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1 |
| m7a.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 12, 18, 24, 30, 36, 42, 48 | 1 |
| m7a.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48, 56, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7a.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 24, 36, 48, 60, 72, 84, 96 | 1 |
| m7a.32xlarge | 128 | 128 | 1 | 4, 6, 8, 10, 12, 14, 16, 32, 48, 64, 80, 96, 112, 128 | 1 |
| m7a.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 48, 72, 96, 120, 144, 168, 192 | 1 |
| m7g.large | 2 | 2 | 1 | 1, 2 | 1 |
| m7g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m7g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m7g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| m7g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| m7gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| m7gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m7gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m7gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| m7gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| m7i.large | 2 | 1 | 2 | 1 | 1, 2 |
| m7i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m7i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m7i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| m7i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m7i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7i.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1, 2 |
| m7i.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m7i-flex.large | 2 | 1 | 2 | 1 | 1, 2 |
| m7i-flex.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m7i-flex.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m7i-flex.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m7i-flex.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| m7i-flex.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m7i-flex.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8a.medium | 1 | 1 | 1 | 1 | 1 |
| m8a.large | 2 | 2 | 1 | 1, 2 | 1 |
| m8a.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m8a.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m8a.4xlarge | 16 | 16 | 1 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1 |
| m8a.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1 |
| m8a.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 12, 18, 24, 30, 36, 42, 48 | 1 |
| m8a.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48, 56, 64 | 1 |
| m8a.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 24, 36, 48, 60, 72, 84, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8a.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 48, 72, 96, 120, 144, 168, 192 | 1 |
| m8azn.medium | 1 | 1 | 1 | 1 | 1 |
| m8azn.large | 2 | 2 | 1 | 1, 2 | 1 |
| m8azn.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m8azn.3xlarge | 12 | 12 | 1 | 1, 2, 4, 6, 8, 10, 12 | 1 |
| m8azn.6xlarge | 24 | 24 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24 | 1 |
| m8azn.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48 | 1 |
| m8azn.24xlarge | 96 | 96 | 1 | 4, 6, 8, 10, 12, 14, 16, 32, 48, 64, 80, 96 | 1 |
| m8g.large | 2 | 2 | 1 | 1, 2 | 1 |
| m8g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m8g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m8g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| m8g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| m8gb.large | 2 | 2 | 1 | 1, 2 | 1 |
| m8gb.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m8gb.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m8gb.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m8gb.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gb.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gb.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gb.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gb.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| m8gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| m8gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m8gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m8gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m8gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gd.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gd.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| m8gn.large | 2 | 2 | 1 | 1, 2 | 1 |
| m8gn.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m8gn.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m8gn.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m8gn.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gn.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gn.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gn.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8gn.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| m8i.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m8i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| m8i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m8i.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| m8i.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8i.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |
| m8i.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| m8id.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8id.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8id.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8id.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8id.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| m8id.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m8id.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m8id.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8id.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m8id.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8id.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| m8i-flex.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8i-flex.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8i-flex.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m8i-flex.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8i-flex.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8i-flex.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m8i-flex.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m8in.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8in.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8in.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m8in.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8in.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8in.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m8in.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m8in.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8in.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m8in.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8in.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| m8idn.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8idn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8idn.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m8idn.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8idn.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8idn.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m8idn.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m8idn.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8idn.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m8idn.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8idn.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| m8ine.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8ine.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8ine.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m8ine.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8ine.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8ine.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m8ib.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8ib.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8ib.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| m8ib.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8ib.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| m8ib.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8ib.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m8ib.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| m8ib.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8ib.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |
| m8ib.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| m8idb.large | 2 | 1 | 2 | 1 | 1, 2 |
| m8idb.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| m8idb.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8idb.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| m8idb.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| m8idb.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| m8idb.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| m8idb.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8idb.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| m8idb.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m8idb.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| m9g.large | 2 | 2 | 1 | 1, 2 | 1 |
| m9g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m9g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m9g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m9g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| m9g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m9g.16xlarge | 64 | 64 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m9g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m9g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| m9gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| m9gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| m9gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| m9gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| m9gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m9gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |
| m9gd.16xlarge | 64 | 64 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m9gd.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| m9gd.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| t3.nano | 2 | 1 | 2 | 1 | 1, 2 |
| t3.micro | 2 | 1 | 2 | 1 | 1, 2 |
| t3.small | 2 | 1 | 2 | 1 | 1, 2 |
| t3.medium | 2 | 1 | 2 | 1 | 1, 2 |
| t3.large | 2 | 1 | 2 | 1 | 1, 2 |
| t3.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| t3.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| t3a.nano | 2 | 1 | 2 | 1 | 1, 2 |
| t3a.micro | 2 | 1 | 2 | 1 | 1, 2 |
| t3a.small | 2 | 1 | 2 | 1 | 1, 2 |
| t3a.medium | 2 | 1 | 2 | 1 | 1, 2 |
| t3a.large | 2 | 1 | 2 | 1 | 1, 2 |
| t3a.xlarge | 4 | 2 | 2 | 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| t3a.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| t4g.nano | 2 | 2 | 1 | 1, 2 | 1 |
| t4g.micro | 2 | 2 | 1 | 1, 2 | 1 |
| t4g.small | 2 | 2 | 1 | 1, 2 | 1 |
| t4g.medium | 2 | 2 | 1 | 1, 2 | 1 |
| t4g.large | 2 | 2 | 1 | 1, 2 | 1 |
| t4g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| t4g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

Instans komputasi yang dioptimalkan

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c3.large | 2 | 1 | 2 | 1 | 1, 2 |
| c3.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c3.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c3.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c3.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| c4.large | 2 | 1 | 2 | 1 | 1, 2 |
| c4.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c4.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c4.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c4.8xlarge | 36 | 18 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18 | 1, 2 |
| c5.large | 2 | 1 | 2 | 1 | 1, 2 |
| c5.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| c5.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| c5.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| c5.9xlarge | 36 | 18 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18 | 1, 2 |
| c5.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c5.18xlarge | 72 | 36 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36 | 1, 2 |
| c5.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| c5a.large | 2 | 1 | 2 | 1 | 1, 2 |
| c5a.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c5a.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c5a.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 8 | 1, 2 |
| c5a.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 8, 12, 16 | 1, 2 |
| c5a.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 8, 12, 16, 20, 24 | 1, 2 |
| c5a.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c5a.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48 | 1, 2 |
| c5ad.large | 2 | 1 | 2 | 1 | 1, 2 |
| c5ad.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c5ad.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c5ad.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 8 | 1, 2 |
| c5ad.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 8, 12, 16 | 1, 2 |
| c5ad.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 8, 12, 16, 20, 24 | 1, 2 |
| c5ad.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1, 2 |
| c5ad.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48 | 1, 2 |
| c5d.large | 2 | 1 | 2 | 1 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c5d.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| c5d.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| c5d.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| c5d.9xlarge | 36 | 18 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18 | 1, 2 |
| c5d.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| c5d.18xlarge | 72 | 36 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36 | 1, 2 |
| c5d.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| c5n.large | 2 | 1 | 2 | 1 | 1, 2 |
| c5n.xlarge | 4 | 2 | 2 | 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c5n.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| c5n.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| c5n.9xlarge | 36 | 18 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18 | 1, 2 |
| c5n.18xlarge | 72 | 36 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36 | 1, 2 |
| c6a.large | 2 | 1 | 2 | 1 | 1, 2 |
| c6a.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c6a.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c6a.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c6a.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| c6a.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24 | 1, 2 |
| c6a.16xlarge | 64 | 32 | 2 | 4, 6, 8, 10, 12, 14, 16, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6a.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 32, 48 | 1, 2 |
| c6a.32xlarge | 128 | 64 | 2 | 8, 12, 16, 20, 24, 28, 32, 64 | 1, 2 |
| c6a.48xlarge | 192 | 96 | 2 | 8, 12, 16, 20, 24, 28, 32, 64, 96 | 1, 2 |
| c6g.large | 2 | 2 | 1 | 1, 2 | 1 |
| c6g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c6g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c6g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c6g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| c6gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| c6gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c6gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c6gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| c6gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| c6gn.medium | 1 | 1 | 1 | 1 | 1 |
| c6gn.large | 2 | 2 | 1 | 1, 2 | 1 |
| c6gn.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c6gn.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6gn.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c6gn.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| c6gn.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6gn.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| c6i.large | 2 | 1 | 2 | 1 | 1, 2 |
| c6i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c6i.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| c6i.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| c6i.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6i.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| c6i.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| c6i.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| c6i.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| c6id.large | 2 | 1 | 2 | 1 | 1, 2 |
| c6id.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6id.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| c6id.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| c6id.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| c6id.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| c6id.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| c6id.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6id.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| c6in.large | 2 | 1 | 2 | 1 | 1, 2 |
| c6in.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c6in.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c6in.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c6in.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6in.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| c6in.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| c6in.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c6in.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| c7a.large | 2 | 2 | 1 | 1, 2 | 1 |
| c7a.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c7a.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c7a.4xlarge | 16 | 16 | 1 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1 |
| c7a.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1 |
| c7a.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 12, 18, 24, 30, 36, 42, 48 | 1 |
| c7a.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48, 56, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7a.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 24, 36, 48, 60, 72, 84, 96 | 1 |
| c7a.32xlarge | 128 | 128 | 1 | 4, 6, 8, 10, 12, 14, 16, 32, 48, 64, 80, 96, 112, 128 | 1 |
| c7a.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 48, 72, 96, 120, 144, 168, 192 | 1 |
| c7g.large | 2 | 2 | 1 | 1, 2 | 1 |
| c7g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c7g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c7g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| c7g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| c7gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| c7gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c7gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c7gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| c7gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| c7gn.large | 2 | 2 | 1 | 1, 2 | 1 |
| c7gn.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c7gn.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7gn.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c7gn.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| c7gn.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7gn.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| c7i.large | 2 | 1 | 2 | 1 | 1, 2 |
| c7i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c7i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c7i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| c7i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| c7i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7i.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1, 2 |
| c7i.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c7i-flex.large | 2 | 1 | 2 | 1 | 1, 2 |
| c7i-flex.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c7i-flex.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c7i-flex.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c7i-flex.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| c7i-flex.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| c7i-flex.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8a.medium | 1 | 1 | 1 | 1 | 1 |
| c8a.large | 2 | 2 | 1 | 1, 2 | 1 |
| c8a.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c8a.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c8a.4xlarge | 16 | 16 | 1 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1 |
| c8a.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1 |
| c8a.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 12, 18, 24, 30, 36, 42, 48 | 1 |
| c8a.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48, 56, 64 | 1 |
| c8a.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 24, 36, 48, 60, 72, 84, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8a.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 48, 72, 96, 120, 144, 168, 192 | 1 |
| c8g.large | 2 | 2 | 1 | 1, 2 | 1 |
| c8g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c8g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c8g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c8g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| c8gb.large | 2 | 2 | 1 | 1, 2 | 1 |
| c8gb.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c8gb.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c8gb.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c8gb.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gb.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gb.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gb.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gb.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| c8gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| c8gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c8gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c8gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c8gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gd.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gd.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| c8gn.large | 2 | 2 | 1 | 1, 2 | 1 |
| c8gn.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c8gn.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c8gn.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c8gn.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gn.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gn.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gn.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8gn.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| c8i.large | 2 | 1 | 2 | 1 | 1, 2 |
| c8i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c8i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c8i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c8i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| c8i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| c8i.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| c8i.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8i.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |
| c8i.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| c8id.large | 2 | 1 | 2 | 1 | 1, 2 |
| c8id.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c8id.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8id.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c8id.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| c8id.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| c8id.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| c8id.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8id.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| c8id.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8id.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| c8i-flex.large | 2 | 1 | 2 | 1 | 1, 2 |
| c8i-flex.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c8i-flex.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c8i-flex.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c8i-flex.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8i-flex.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| c8i-flex.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| c8in.large | 2 | 1 | 2 | 1 | 1, 2 |
| c8in.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c8in.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c8in.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c8in.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8in.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| c8in.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| c8in.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8in.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| c8in.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8in.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| c8ine.large | 2 | 1 | 2 | 1 | 1, 2 |
| c8ine.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c8ine.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c8ine.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c8ine.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8ine.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| c8ib.large | 2 | 1 | 2 | 1 | 1, 2 |
| c8ib.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| c8ib.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| c8ib.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| c8ib.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| c8ib.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8ib.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| c8ib.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| c8ib.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c8ib.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |
| c8ib.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| c9g.large | 2 | 2 | 1 | 1, 2 | 1 |
| c9g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c9g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c9g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c9g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| c9g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c9g.16xlarge | 64 | 64 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c9g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c9g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| c9gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| c9gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| c9gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| c9gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| c9gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c9gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |
| c9gd.16xlarge | 64 | 64 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c9gd.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| c9gd.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |

Instans memori yang dioptimalkan

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r3.large | 2 | 1 | 2 | 1 | 1, 2 |
| r3.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r3.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r3.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r3.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r4.large | 2 | 1 | 2 | 1 | 1, 2 |
| r4.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r4.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r4.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r4.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| r4.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5.large | 2 | 1 | 2 | 1 | 1, 2 |
| r5.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| r5.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r5.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r5.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r5.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r5.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| r5a.large | 2 | 1 | 2 | 1 | 1, 2 |
| r5a.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| r5a.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r5a.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r5a.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r5a.12xlarge | 48 | 24 | 2 | 6, 12, 18, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r5a.16xlarge | 64 | 32 | 2 | 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5a.24xlarge | 96 | 48 | 2 | 12, 18, 24, 36, 48 | 1, 2 |
| r5ad.large | 2 | 1 | 2 | 1 | 1, 2 |
| r5ad.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| r5ad.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r5ad.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r5ad.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r5ad.12xlarge | 48 | 24 | 2 | 6, 12, 18, 24 | 1, 2 |
| r5ad.16xlarge | 64 | 32 | 2 | 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5ad.24xlarge | 96 | 48 | 2 | 12, 18, 24, 36, 48 | 1, 2 |
| r5b.large | 2 | 1 | 2 | 1 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r5b.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r5b.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r5b.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r5b.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r5b.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| r5b.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5b.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| r5d.large | 2 | 1 | 2 | 1 | 1, 2 |
| r5d.xlarge | 4 | 2 | 2 | 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r5d.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r5d.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r5d.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r5d.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| r5d.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5d.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| r5dn.large | 2 | 1 | 2 | 1 | 1, 2 |
| r5dn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r5dn.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r5dn.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r5dn.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r5dn.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| r5dn.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5dn.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| r5n.large | 2 | 1 | 2 | 1 | 1, 2 |
| r5n.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r5n.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r5n.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r5n.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r5n.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| r5n.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r5n.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| r6a.large | 2 | 1 | 2 | 1 | 1, 2 |
| r6a.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r6a.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r6a.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6a.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r6a.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24 | 1, 2 |
| r6a.16xlarge | 64 | 32 | 2 | 4, 6, 8, 10, 12, 14, 16, 32 | 1, 2 |
| r6a.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 32, 48 | 1, 2 |
| r6a.32xlarge | 128 | 64 | 2 | 8, 12, 16, 20, 24, 28, 32, 64 | 1, 2 |
| r6a.48xlarge | 192 | 96 | 2 | 8, 12, 16, 20, 24, 28, 32, 64, 96 | 1, 2 |
| r6g.large | 2 | 2 | 1 | 1, 2 | 1 |
| r6g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r6g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r6g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| r6g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| r6gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| r6gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r6gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| r6gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| r6gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| r6i.large | 2 | 1 | 2 | 1 | 1, 2 |
| r6i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r6i.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r6i.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r6i.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6i.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| r6i.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r6i.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| r6i.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r6id.large | 2 | 1 | 2 | 1 | 1, 2 |
| r6id.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6id.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| r6id.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| r6id.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| r6id.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| r6id.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| r6id.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6id.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r6idn.large | 2 | 1 | 2 | 1 | 1, 2 |
| r6idn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r6idn.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r6idn.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r6idn.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6idn.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r6idn.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r6idn.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6idn.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r6in.large | 2 | 1 | 2 | 1 | 1, 2 |
| r6in.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r6in.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r6in.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r6in.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6in.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r6in.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r6in.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r6in.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r7a.large | 2 | 2 | 1 | 1, 2 | 1 |
| r7a.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r7a.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r7a.4xlarge | 16 | 16 | 1 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1 |
| r7a.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1 |
| r7a.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 12, 18, 24, 30, 36, 42, 48 | 1 |
| r7a.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48, 56, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7a.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 24, 36, 48, 60, 72, 84, 96 | 1 |
| r7a.32xlarge | 128 | 128 | 1 | 4, 6, 8, 10, 12, 14, 16, 32, 48, 64, 80, 96, 112, 128 | 1 |
| r7a.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 48, 72, 96, 120, 144, 168, 192 | 1 |
| r7g.large | 2 | 2 | 1 | 1, 2 | 1 |
| r7g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r7g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r7g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| r7g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| r7gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| r7gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r7gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| r7gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| r7gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| r7i.large | 2 | 1 | 2 | 1 | 1, 2 |
| r7i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r7i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r7i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| r7i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r7i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7i.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1, 2 |
| r7i.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7iz.large | 2 | 1 | 2 | 1 | 1, 2 |
| r7iz.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r7iz.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r7iz.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r7iz.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| r7iz.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r7iz.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r7iz.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r8a.medium | 1 | 1 | 1 | 1 | 1 |
| r8a.large | 2 | 2 | 1 | 1, 2 | 1 |
| r8a.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r8a.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r8a.4xlarge | 16 | 16 | 1 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1 |
| r8a.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1 |
| r8a.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 12, 18, 24, 30, 36, 42, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8a.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48, 56, 64 | 1 |
| r8a.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 24, 36, 48, 60, 72, 84, 96 | 1 |
| r8a.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 48, 72, 96, 120, 144, 168, 192 | 1 |
| r8g.large | 2 | 2 | 1 | 1, 2 | 1 |
| r8g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r8g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r8g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| r8g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| r8gb.large | 2 | 2 | 1 | 1, 2 | 1 |
| r8gb.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r8gb.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r8gb.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| r8gb.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gb.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gb.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gb.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gb.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| r8gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| r8gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r8gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r8gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| r8gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gd.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gd.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| r8gn.large | 2 | 2 | 1 | 1, 2 | 1 |
| r8gn.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r8gn.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r8gn.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| r8gn.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gn.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gn.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gn.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8gn.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| r8i.large | 2 | 1 | 2 | 1 | 1, 2 |
| r8i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r8i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r8i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r8i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| r8i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r8i.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| r8i.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8i.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |
| r8i.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| r8id.large | 2 | 1 | 2 | 1 | 1, 2 |
| r8id.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r8id.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8id.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r8id.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| r8id.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r8id.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r8id.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8id.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r8id.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8id.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| r8i-flex.large | 2 | 1 | 2 | 1 | 1, 2 |
| r8i-flex.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r8i-flex.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r8i-flex.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r8i-flex.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8i-flex.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r8i-flex.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r8in.large | 2 | 1 | 2 | 1 | 1, 2 |
| r8in.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r8in.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r8in.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r8in.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8in.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r8in.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r8in.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8in.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r8in.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8in.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| r8idn.large | 2 | 1 | 2 | 1 | 1, 2 |
| r8idn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r8idn.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r8idn.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r8idn.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8idn.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r8idn.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r8idn.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8idn.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r8idn.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8idn.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| r8ib.large | 2 | 1 | 2 | 1 | 1, 2 |
| r8ib.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r8ib.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r8ib.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r8ib.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8ib.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r8ib.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r8ib.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8ib.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r8ib.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8ib.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| r8idb.large | 2 | 1 | 2 | 1 | 1, 2 |
| r8idb.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| r8idb.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| r8idb.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| r8idb.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8idb.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| r8idb.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| r8idb.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8idb.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| r8idb.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r8idb.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| r9g.large | 2 | 2 | 1 | 1, 2 | 1 |
| r9g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r9g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r9g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r9g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| r9g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r9g.16xlarge | 64 | 64 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r9g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r9g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| r9gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| r9gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| r9gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| r9gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| r9gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r9gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |
| r9gd.16xlarge | 64 | 64 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r9gd.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| r9gd.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| u-3tb1.56xlarge | 224 | 112 | 2 | 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 68, 72, 76, 80, 84, 88, 92, 96, 100, 104, 108, 112 | 1, 2 |
| u-6tb1.56xlarge | 224 | 224 | 1 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u-6tb1.112xlarge | 448 | 224 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224 | 1, 2 |
| u-9tb1.112xlarge | 448 | 224 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u-12tb1.112xlarge | 448 | 224 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224 | 1, 2 |
| u-18tb1.112xlarge | 448 | 224 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u-24tb1.112xlarge | 448 | 224 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u7i-6tb.112xlarge | 448 | 224 | 2 | 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 68, 72, 76, 80, 84, 88, 92, 96, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 144, 148, 152, 156, 160, 164, 168, 172, 176, 180, 184, 188, 192, 196, 200, 204, 208, 212, 216, 220, 224 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u7i-8tb.112xlarge | 448 | 224 | 2 | 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 68, 72, 76, 80, 84, 88, 92, 96, 100, 104, 108, 112, 116, 120, 124, 128, 132, 136, 140, 144, 148, 152, 156, 160, 164, 168, 172, 176, 180, 184, 188, 192, 196, 200, 204, 208, 212, 216, 220, 224 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u7i-12tb.224xlarge | 896 | 448 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224, 232, 240, 248, 256, 264, 272, 280, 288, 296, 304, 312, 320, 328, 336, 344, 352, 360, 368, 376, 384, 392, 400, 408, 416, 424, 432, 440, 448 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u7in-16tb.224xlarge | 896 | 448 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224, 232, 240, 248, 256, 264, 272, 280, 288, 296, 304, 312, 320, 328, 336, 344, 352, 360, 368, 376, 384, 392, 400, 408, 416, 424, 432, 440, 448 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u7in-24tb.224xlarge | 896 | 448 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224, 232, 240, 248, 256, 264, 272, 280, 288, 296, 304, 312, 320, 328, 336, 344, 352, 360, 368, 376, 384, 392, 400, 408, 416, 424, 432, 440, 448 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u7in-32tb.224xlarge | 896 | 448 | 2 | 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120, 128, 136, 144, 152, 160, 168, 176, 184, 192, 200, 208, 216, 224, 232, 240, 248, 256, 264, 272, 280, 288, 296, 304, 312, 320, 328, 336, 344, 352, 360, 368, 376, 384, 392, 400, 408, 416, 424, 432, 440, 448 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| u7inh-32tb.480xlarge | 1920 | 960 | 2 | 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, 240, 256, 272, 288, 304, 320, 336, 352, 368, 384, 400, 416, 432, 448, 464, 480, 496, 512, 528, 544, 560, 576, 592, 608, 624, 640, 656, 672, 688, 704, 720, 736, 752, 768, 784, 800, 816, 832, 848, 864, 880, 896, 912, 928, 944, 960 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x1.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| x1.32xlarge | 128 | 64 | 2 | 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64 | 1, 2 |
| x1e.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| x1e.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| x1e.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| x1e.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| x1e.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x1e.32xlarge | 128 | 64 | 2 | 4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64 | 1, 2 |
| x2gd.large | 2 | 2 | 1 | 1, 2 | 1 |
| x2gd.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| x2gd.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| x2gd.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| x2gd.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x2gd.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x2gd.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| x2idn.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x2idn.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| x2idn.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| x2iedn.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| x2iedn.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| x2iedn.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| x2iedn.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x2iedn.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| x2iedn.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| x2iedn.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| x2iezn.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| x2iezn.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| x2iezn.6xlarge | 24 | 12 | 2 | 2, 4, 6, 8, 10, 12 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x2iezn.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| x2iezn.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| x8g.large | 2 | 2 | 1 | 1, 2 | 1 |
| x8g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| x8g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| x8g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| x8g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x8g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x8g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x8g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x8g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| x8aedz.large | 2 | 2 | 1 | 1, 2 | 1 |
| x8aedz.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| x8aedz.3xlarge | 12 | 12 | 1 | 1, 2, 4, 6, 8, 10, 12 | 1 |
| x8aedz.6xlarge | 24 | 24 | 1 | 1, 2, 3, 4, 8, 12, 16, 20, 24 | 1 |
| x8aedz.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 40, 48 | 1 |
| x8aedz.24xlarge | 96 | 96 | 1 | 4, 6, 8, 10, 12, 14, 16, 32, 48, 64, 80, 96 | 1 |
| x8i.large | 2 | 1 | 2 | 1 | 1, 2 |
| x8i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x8i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| x8i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| x8i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| x8i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| x8i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x8i.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| x8i.32xlarge | 128 | 64 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| x8i.48xlarge | 192 | 96 | 2 | 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57, 60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90, 93, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| x8i.64xlarge | 256 | 128 | 2 | 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64, 68, 72, 76, 80, 84, 88, 92, 96, 100, 104, 108, 112, 116, 120, 124, 128 | 1, 2 |
| x8i.96xlarge | 384 | 192 | 2 | 12, 18, 24, 30, 36, 42, 48, 54, 60, 66, 72, 78, 84, 90, 96, 102, 108, 114, 120, 126, 132, 138, 144, 150, 156, 162, 168, 174, 180, 186, 192 | 1, 2 |
| z1d.large | 2 | 1 | 2 | 1 | 1, 2 |
| z1d.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| z1d.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| z1d.3xlarge | 12 | 6 | 2 | 2, 4, 6 | 1, 2 |
| z1d.6xlarge | 24 | 12 | 2 | 2, 4, 6, 8, 10, 12 | 1, 2 |
| z1d.12xlarge | 48 | 24 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |

Instans penyimpanan yang dioptimalkan

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| d2.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| d2.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| d2.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| d2.8xlarge | 36 | 18 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18 | 1, 2 |
| d3.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| d3.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| d3.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| d3.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| d3en.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| d3en.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| d3en.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| d3en.6xlarge | 24 | 12 | 2 | 2, 4, 6, 8, 10, 12 | 1, 2 |
| d3en.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| d3en.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| h1.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| h1.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| h1.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| h1.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| i2.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| i2.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| i2.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| i2.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| i3.large | 2 | 1 | 2 | 1 | 1, 2 |
| i3.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| i3.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| i3.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| i3.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i3.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| i3en.large | 2 | 1 | 2 | 1 | 1, 2 |
| i3en.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| i3en.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| i3en.3xlarge | 12 | 6 | 2 | 2, 4, 6 | 1, 2 |
| i3en.6xlarge | 24 | 12 | 2 | 2, 4, 6, 8, 10, 12 | 1, 2 |
| i3en.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| i3en.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| i4g.large | 2 | 2 | 1 | 1, 2 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i4g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| i4g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| i4g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| i4g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i4g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| i4i.large | 2 | 1 | 2 | 1 | 1, 2 |
| i4i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| i4i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| i4i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i4i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| i4i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| i4i.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| i4i.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i4i.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| i7i.large | 2 | 1 | 2 | 1 | 1, 2 |
| i7i.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| i7i.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| i7i.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| i7i.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| i7i.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i7i.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1, 2 |
| i7i.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i7i.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |
| i7ie.large | 2 | 1 | 2 | 1 | 1, 2 |
| i7ie.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| i7ie.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| i7ie.3xlarge | 12 | 6 | 2 | 1, 2, 3, 4, 5, 6 | 1, 2 |
| i7ie.6xlarge | 24 | 12 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i7ie.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |
| i7ie.18xlarge | 72 | 36 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i7ie.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1, 2 |
| i7ie.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |
| i8g.large | 2 | 2 | 1 | 1, 2 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| i8g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| i8g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| i8g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8g.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8g.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8g.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| i8ge.large | 2 | 2 | 1 | 1, 2 | 1 |
| i8ge.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| i8ge.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| i8ge.3xlarge | 12 | 12 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 | 1 |
| i8ge.6xlarge | 24 | 24 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8ge.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8ge.18xlarge | 72 | 72 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8ge.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| i8ge.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| im4gn.large | 2 | 2 | 1 | 1, 2 | 1 |
| im4gn.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| im4gn.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| im4gn.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| im4gn.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| im4gn.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| is4gen.medium | 1 | 1 | 1 | 1 | 1 |
| is4gen.large | 2 | 2 | 1 | 1, 2 | 1 |
| is4gen.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |
| is4gen.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| is4gen.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| is4gen.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

Instans komputasi terakselerasi

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| dl1.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| dl2q.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | |
| f2.6xlarge | 24 | 12 | 2 | 1, 2, 3, 6, 9, 12 | 1, 2 |
| f2.12xlarge | 48 | 24 | 2 | 1, 2, 3, 6, 9, 12, 15, 18, 21, 24 | 1, 2 |
| f2.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 24, 36, 48, 60, 72, 84, 96 | 1, 2 |
| g3.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| g3.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| g3.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| g4ad.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| g4ad.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g4ad.4xlarge | 16 | 8 | 2 | 2, 4, 8 | 1, 2 |
| g4ad.8xlarge | 32 | 16 | 2 | 2, 4, 8, 16 | 1, 2 |
| g4ad.16xlarge | 64 | 32 | 2 | 2, 4, 8, 16, 32 | 1, 2 |
| g4dn.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| g4dn.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| g4dn.4xlarge | 16 | 8 | 2 | 2, 4, 6, 8 | 1, 2 |
| g4dn.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| g4dn.12xlarge | 48 | 24 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24 | 1, 2 |
| g4dn.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| g5g.xlarge | 4 | 4 | 1 | 1, 2, 3, 4 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g5g.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| g5g.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |
| g5g.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g5g.16xlarge | 64 | 64 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64 | 1 |
| g6.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| g6.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| g6.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| g6.8xlarge | 32 | 16 | 2 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g6.12xlarge | 48 | 24 | 2 | 1, 2, 3, 6, 9, 12, 15, 18, 21, 24 | 1, 2 |
| g6.16xlarge | 64 | 32 | 2 | 1, 2, 3, 4, 8, 12, 16, 20, 24, 28, 32 | 1, 2 |
| g6.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 5, 6, 12, 18, 24, 30, 36, 42, 48 | 1, 2 |
| g6.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 24, 36, 48, 60, 72, 84, 96 | 1, 2 |
| g6e.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| g6e.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| g6e.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| g6e.8xlarge | 32 | 16 | 2 | 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| g6e.12xlarge | 48 | 24 | 2 | 3, 6, 9, 12, 15, 18, 21, 24 | 1, 2 |
| g6e.16xlarge | 64 | 32 | 2 | 4, 8, 12, 16, 20, 24, 28, 32 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g6e.24xlarge | 96 | 48 | 2 | 6, 12, 18, 24, 30, 36, 42, 48 | 1, 2 |
| g6e.48xlarge | 192 | 96 | 2 | 12, 24, 36, 48, 60, 72, 84, 96 | 1, 2 |
| g6f.large | 2 | 1 | 2 | 1 | 1, 2 |
| g6f.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| g6f.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| g6f.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| gr6.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| gr6.8xlarge | 32 | 16 | 2 | 1, 2, 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| gr6f.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| g7.2xlarge | 8 | 8 | 1 | 1, 2, 3, 4, 5, 6, 7, 8 | 1 |
| g7.4xlarge | 16 | 16 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g7.8xlarge | 32 | 32 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32 | 1 |
| g7.12xlarge | 48 | 48 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g7.24xlarge | 96 | 96 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g7.48xlarge | 192 | 192 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96, 98, 100, 102, 104, 106, 108, 110, 112, 114, 116, 118, 120, 122, 124, 126, 128, 130, 132, 134, 136, 138, 140, 142, 144, 146, 148, 150, 152, 154, 156, 158, 160, 162, 164, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 166, 168, 170, 172, 174, 176, 178, 180, 182, 184, 186, 188, 190, 192 | |
| g7e.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |
| g7e.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| g7e.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| g7e.12xlarge | 48 | 24 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| g7e.24xlarge | 96 | 48 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48 | 1, 2 |
| g7e.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| inf1.xlarge | 4 | 2 | 2 | 2 | 1, 2 |
| inf1.2xlarge | 8 | 4 | 2 | 2, 4 | 1, 2 |
| inf1.6xlarge | 24 | 12 | 2 | 2, 4, 6, 8, 10, 12 | 1, 2 |
| inf1.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| inf2.xlarge | 4 | 2 | 2 | 1, 2 | 1, 2 |
| inf2.8xlarge | 32 | 16 | 2 | 4, 6, 8, 10, 12, 14, 16 | 1, 2 |
| inf2.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 32, 48 | 1, 2 |
| inf2.48xlarge | 192 | 96 | 2 | 4, 8, 12, 16, 20, 24, 28, 32, 64, 96 | 1, 2 |
| p3.2xlarge | 8 | 4 | 2 | 1, 2, 3, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| p3.8xlarge | 32 | 16 | 2 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16 | 1, 2 |
| p3.16xlarge | 64 | 32 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32 | 1, 2 |
| p3dn.24xlarge | 96 | 48 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| p4d.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| p4de.24xlarge | 96 | 48 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48 | 1, 2 |
| p5.4xlarge | 16 | 8 | 2 | 1, 2, 3, 4, 5, 6, 7, 8 | 1, 2 |
| p5.48xlarge | 192 | 96 | 2 | 12, 24, 36, 48, 60, 72, 84, 96 | 1, 2 |
| p5e.48xlarge | 192 | 96 | 2 | 12, 24, 36, 48, 60, 72, 84, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| p5en.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |
| p6-b200.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| p6-b300.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| p6e-gb200.36xlarge | 144 | 144 | 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, | 1 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| | | | | 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136 , 137, 138, 139, 140, 141, 142, 143, 144 | |
| `trn1.2xlarge` | 8 | 4 | 2 | 2, 4 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| trn1.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| trn1n.32xlarge | 128 | 64 | 2 | 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1, 2 |
| trn2.3xlarge | 12 | 6 | 2 | 1, 2, 3, 4, 5, 6 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| trn2.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |
| trn2u.48xlarge | 192 | 96 | 2 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96 | 1, 2 |

| Jenis instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| vt1.3xlarge | 12 | 6 | 2 | 6 | 1, 2 |
| vt1.6xlarge | 24 | 12 | 2 | 6, 12 | 1, 2 |
| vt1.24xlarge | 96 | 48 | 2 | 6, 12, 48 | 1, 2 |

High-performance instans komputasi

| Tipe instans | vCPU default | Inti CPU default | Thread default per inti | Inti CPU yang valid | Thread valid per inti |
|---|---|---|---|---|---|
| hpc6id.32xlarge | 64 | 64 | 1 | 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64 | 1 |
| hpc8a.96xlarge | 192 | 192 | 1 | 24, 48, 72, 96, 120, 144, 168, 192 | 1 |

# Tentukan opsi CPU untuk instans Amazon EC2

Anda dapat menentukan opsi CPU selama atau setelah peluncuran instance.

Tugas

- Nonaktifkan multithreading simultan
- Tentukan jumlah kustom vCPU saat peluncuran
- Tentukan jumlah kustom vCPU dalam templat peluncuran
- Mengubah opsi CPU untuk instans EC2 Anda

## Nonaktifkan multithreading simultan

Untuk menonaktifkan multithreading simultan (SMT), juga dikenal sebagai hyper-threading, tentukan 1 thread per inti.

### Console

Untuk menonaktifkan SMT selama peluncuran instance

1. Ikuti prosedur Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol dan konfigurasikan instans Anda sesuai kebutuhan.
2. Perluas detail lanj utan, dan pilih kotak centang T entukan opsi CPU.
3. Untuk Jumlah inti, pilih jumlah inti CPU yang diperlukan. Dalam contoh ini, untuk menentukan jumlah inti CPU default untuk instans `r5.4xlarge`, pilih 8.
4. Untuk menonaktifkan SMT, untuk Th read per core, pilih 1.
5. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

### AWS CLI

Untuk menonaktifkan SMT selama peluncuran instance

Gunakan AWS CLI perintah run-instance dan tentukan nilai for 1 `ThreadsPerCore` untuk parameter. `--cpu-options` Untuk `CoreCount`, tentukan jumlah inti CPU. Dalam contoh ini, untuk menentukan jumlah inti CPU default untuk instans `r7i.4xlarge`, tentukan nilai 8.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type r7i.4xlarge \
    --cpu-options "CoreCount=8,ThreadsPerCore=1" \
    --key-name my-key-pair
```

PowerShell

Untuk menonaktifkan SMT selama peluncuran instance

Gunakan New-EC2Instance perintah dan tentukan nilai 1 for `ThreadsPerCore` untuk `-CpuOptions` parameter. Untuk `CoreCount`, tentukan jumlah inti CPU. Dalam contoh ini, untuk menentukan jumlah inti CPU default untuk instans `r7i.4xlarge`, tentukan nilai 8.

```
New-EC2Instance `
    -ImageId 'ami-0abcdef1234567890' `
    -InstanceType 'r7i.4xlarge' `
    -CpuOptions @{CoreCount=8; ThreadsPerCore=1} `
    -KeyName 'my-key-pair'
```

> **Note**
>
> Untuk menonaktifkan SMT untuk instance yang ada, ikuti proses yang ditunjukkan diMengubah opsi CPU untuk instans EC2 Anda, dan ubah jumlah utas yang dijalankan per inti menjadi1.

Tentukan jumlah kustom vCPU saat peluncuran

Anda dapat menyesuaikan jumlah inti CPU dan utas per inti saat meluncurkan instans dari konsol EC2 atau AWS CLI. Contoh di bagian ini menggunakan jenis `r5.4xlarge` instance, yang memiliki pengaturan default berikut:

- Inti CPU: 8
- Utas per inti: 2

Instans diluncurkan dengan jumlah maksimum vCPU yang tersedia untuk jenis instans secara default. Untuk jenis contoh ini, itu total 16 vCPU (8 core masing-masing menjalankan 2 utas). Untuk informasi selengkapnya tentang jenis instance ini, lihatInstans memori yang dioptimalkan.

Contoh berikut meluncurkan `r5.4xlarge` instance dengan 4 vCPU.

Console

Untuk menentukan jumlah kustom vCPU selama peluncuran instans

1. Ikuti prosedur Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol dan konfigurasikan instans Anda sesuai kebutuhan.
2. Perluas detail lanj utan, dan pilih kotak centang T entukan opsi CPU.
3. Untuk mendapatkan 4 vCPU, tentukan 2 inti CPU dan 2 utas per inti, sebagai berikut:
   - Untuk jumlah inti, pilih 2.
   - Untuk Thread per inti, pilih 2.
4. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

AWS CLI

Untuk menentukan jumlah kustom vCPU selama peluncuran instans

Gunakan AWS CLI perintah run-instance dan tentukan jumlah inti CPU dan jumlah utas dalam parameter. `--cpu-options` Anda dapat menentukan 2 inti CPU dan 2 utas per inti untuk mendapatkan 4 vCPU.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type r7i.4xlarge \
    --cpu-options "CoreCount=2,ThreadsPerCore=2" \
    --key-name my-key-pair
```

Atau, tentukan 4 inti CPU dan 1 utas per inti (nonaktifkan SMT) untuk mendapatkan 4 vCPU:

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type r7i.4xlarge \
```

```
    --cpu-options "CoreCount=4,ThreadsPerCore=1" \
    --key-name my-key-pair
```

PowerShell

Untuk menentukan jumlah kustom vCPU selama peluncuran instans

Gunakan New-EC2Instance perintah dan tentukan jumlah inti CPU dan jumlah utas dalam `-CpuOptions` parameter. Anda dapat menentukan 2 inti CPU dan 2 utas per inti untuk mendapatkan 4 vCPU.

```
New-EC2Instance `
    -ImageId 'ami-0abcdef1234567890' `
    -InstanceType 'r7i.4xlarge' `
    -CpuOptions @{CoreCount=2; ThreadsPerCore=2} `
    -KeyName 'my-key-pair'
```

Atau, tentukan 4 inti CPU dan 1 utas per inti (nonaktifkan SMT) untuk mendapatkan 4 vCPU:

```
New-EC2Instance `
    -ImageId 'ami-0abcdef1234567890' `
    -InstanceType 'r7i.4xlarge' `
    -CpuOptions @{CoreCount=4; ThreadsPerCore=1} `
    -KeyName 'my-key-pair'
```

Tentukan jumlah kustom vCPU dalam templat peluncuran

Anda dapat menyesuaikan jumlah inti CPU dan thread per inti untuk instans dalam templat peluncuran. Contoh di bagian ini menggunakan jenis `r5.4xlarge` instance, yang memiliki pengaturan default berikut:

- Inti CPU: 8
- Utas per inti: 2

Instans diluncurkan dengan jumlah maksimum vCPU yang tersedia untuk jenis instans secara default. Untuk jenis contoh ini, itu total 16 vCPU (8 core masing-masing menjalankan 2 utas). Untuk informasi selengkapnya tentang jenis instance ini, lihatInstans memori yang dioptimalkan.

Contoh berikut membuat template peluncuran yang menentukan konfigurasi untuk `r5.4xlarge` instance dengan 4 vCPU.

Console

Untuk menentukan jumlah kustom vCPU dalam templat peluncuran

1. Ikuti prosedur Buat template peluncuran dengan menentukan parameter dan konfigurasikan templat peluncuran Anda sesuai kebutuhan.
2. Perluas detail lanj utan, dan pilih kotak centang T entukan opsi CPU.
3. Untuk mendapatkan 4 vCPU, tentukan 2 inti CPU dan 2 utas per inti, sebagai berikut:
   - Untuk jumlah inti, pilih 2.
   - Untuk Thread per inti, pilih 2.
4. Di panel Ringkasan, tinjau konfigurasi instans Anda dan pilih Buat templat peluncuran. Untuk informasi selengkapnya, lihat Simpan parameter peluncuran instans di templat peluncuran Amazon EC2.

AWS CLI

Untuk menentukan jumlah kustom vCPU dalam templat peluncuran

Gunakan AWS CLI perintah create-launch-template dan tentukan jumlah inti CPU dan jumlah utas dalam parameter. `CpuOptions` Anda dapat menentukan 2 inti CPU dan 2 utas per inti untuk mendapatkan 4 vCPU.

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForCPUOptions \
    --version-description CPUOptionsVersion1 \
    --launch-template-data file://template-data.json
```

Berikut ini adalah contoh file JSON yang berisi data templat peluncuran, yang mencakup opsi CPU, untuk konfigurasi instans untuk contoh ini.

```
{
    "NetworkInterfaces": [{
        "AssociatePublicIpAddress": true,
        "DeviceIndex": 0,
        "Ipv6AddressCount": 1,
        "SubnetId": "subnet-0abcdef1234567890"
    }],
    "ImageId": "ami-0abcdef1234567890",
```

```
    "InstanceType": "r5.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
    }],
    "CpuOptions": {
        "CoreCount":2,
        "ThreadsPerCore":2
    }
}
```

Atau, tentukan 4 inti CPU dan 1 utas per inti (nonaktifkan SMT) untuk mendapatkan 4 vCPU:

```
{
    "NetworkInterfaces": [{
        "AssociatePublicIpAddress": true,
        "DeviceIndex": 0,
        "Ipv6AddressCount": 1,
        "SubnetId": "subnet-0abcdef1234567890"
    }],
    "ImageId": "ami-0abcdef1234567890",
    "InstanceType": "r5.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
    }],
    "CpuOptions": {
        "CoreCount":4,
        "ThreadsPerCore":1
    }
}
```

PowerShell

Untuk menentukan jumlah kustom vCPU dalam templat peluncuran

Gunakan New-EC2LaunchTemplate.

```
New-EC2LaunchTemplate `
    -LaunchTemplateName 'TemplateForCPUOptions' `
    -VersionDescription 'CPUOptionsVersion1' `
    -LaunchTemplateData (Get-Content -Path 'template-data.json' | ConvertFrom-Json)
```

## Mengubah opsi CPU untuk instans EC2 Anda

Karena kebutuhan Anda berubah dari waktu ke waktu, Anda mungkin ingin mengubah konfigurasi opsi CPU untuk instance yang ada. Setiap thread yang berjalan pada instance Anda dikenal sebagai CPU virtual (vCPU). Anda dapat mengubah jumlah vCPU yang berjalan untuk instans yang ada di konsol, API AWS CLI, atau SDK Amazon EC2. Status instance harus ada `Stopped` sebelum Anda dapat membuat perubahan ini.

Untuk melihat langkah konsol atau baris perintah, pilih tab yang cocok dengan lingkungan Anda. Untuk informasi permintaan dan respons API, lihat ModifyInstanceCpuOptions di Refer ensi API Amazon EC2.

### Console

Ikuti prosedur ini untuk mengubah jumlah vCPU aktif untuk instance Anda dari Konsol Manajemen AWS.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans. Ini membuka daftar instance yang ditentukan untuk saat ini Wilayah AWS.
3. Pilih instance dari daftar Instans. Atau, Anda dapat memilih tautan instance untuk membuka halaman detail instance.
4. Jika instance sedang berjalan, Anda harus menghentikannya sebelum melanjutkan. Pilih Hentikan instance dari menu Status Instance.
5. Untuk mengubah konfigurasi vCPU, pilih Ubah opsi CPU dari Pengaturan instans di menu Tind akan. Ini membuka halaman Ubah opsi CPU.
6. Pilih salah satu opsi CPU berikut untuk mengubah konfigurasi instans Anda.

   Gunakan opsi CPU default

   Opsi ini mengatur ulang instance Anda ke jumlah default vCPU untuk jenis instans. Defaultnya adalah menjalankan semua utas untuk semua inti CPU.

Tentukan opsi CPU

Opsi ini mengaktifkan konfigurasi jumlah vCPU yang berjalan pada instans Anda.

7. Jika Anda memilih Tentukan opsi CPU, bidang vCPU Aktif ditampilkan.

   - Gunakan pemilih pertama untuk mengkonfigurasi jumlah utas untuk setiap inti CPU. Untuk menonaktifkan multithreading simultan, pilih. 1
   - Gunakan pemilih kedua untuk mengonfigurasi jumlah CPU yang berjalan pada instans Anda.

   Bidang berikut diperbarui secara dinamis saat Anda membuat perubahan pada pemilih opsi CPU.

   - Active vCPU: Jumlah core CPU dikalikan dengan utas per inti, berdasarkan pilihan yang Anda buat. Misalnya, jika Anda memilih 2 utas dan 4 core, itu akan sama dengan 8 vCPU.
   - Total vCPU: Jumlah maksimum vCPU untuk jenis instance. Misalnya, untuk tipe `m6i.4xlarge` instance, ini adalah 16 vCPU (8 core masing-masing menjalankan 2 utas).
8. Untuk menerapkan pembaruan Anda, pilih U bah.

AWS CLI

Ikuti prosedur ini untuk mengubah jumlah vCPU aktif untuk instance Anda dari AWS CLI.

Gunakan perintah modify-instance-cpu-options dan tentukan jumlah inti CPU yang berjalan di `--core-count` parameter, dan jumlah utas yang berjalan per inti dalam parameter. `--threads-per-core`

Contoh berikut menunjukkan dua kemungkinan konfigurasi pada tipe `m6i.4xlarge` instance untuk menjalankan 8 vCPU pada instance yang ditentukan. Default untuk jenis instance ini adalah 16 vCPU (8 core masing-masing menjalankan 2 utas).

Contoh 1: J alankan 4 inti CPU dengan 2 utas per inti, dengan total 8 vCPU.

```
aws ec2 modify-instance-cpu-options \
    --instance-id i-1234567890abcdef0 \

    --core-count=4 \
    --threads-per-core=2
```

Contoh 2: N onaktifkan multi-threading simultan dengan mengubah jumlah utas yang berjalan per inti menjadi. 1 Konfigurasi yang dihasilkan juga menjalankan total 8 vCPU (8 inti CPU dengan 1 utas per inti).

```
aws ec2 modify-instance-cpu-options \
    --instance-id 1234567890abcdef0 \
    --core-count=8 \
    --threads-per-core=1
```

PowerShell

Untuk mengubah jumlah vCPU aktif untuk sebuah instance

Gunakan Edit-EC2InstanceCpuOption cmdlet dan tentukan jumlah inti CPU yang berjalan di `-CoreCount` parameter, dan jumlah utas yang berjalan per inti dalam parameter. `ThreadsPerCore`

Contoh 1: J alankan 4 inti CPU dengan 2 utas per inti, dengan total 8 vCPU.

```
Edit-EC2InstanceCpuOption `
    -InstanceId 'i-1234567890abcdef0' `
    -CoreCount 4 `
    -ThreadsPerCore 2
```

Contoh 2: N onaktifkan multi-threading simultan dengan mengubah jumlah utas yang berjalan per inti menjadi. 1 Konfigurasi yang dihasilkan juga menjalankan total 8 vCPU (8 inti CPU dengan 1 utas per inti).

```
Edit-EC2InstanceCpuOption `
    -InstanceId 'i-1234567890abcdef0' `
    -CoreCount 8 `
    -ThreadsPerCore 1
```

## Melihat utas dan inti CPU untuk instans Amazon EC2

Anda dapat melihat opsi CPU untuk instance yang ada dengan menjelaskan instance tersebut.

Console

Untuk melihat opsi CPU untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi kiri, pilih Instans, kemudian pilih instans.
3. Di tab Detail, pada Host dan grup penempatan, temukan Jumlah vCPU.

AWS CLI

Untuk melihat opsi CPU untuk sebuah instance

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[].Instances[].CpuOptions
```

Berikut ini adalah output contoh. `CoreCount`Bidang menunjukkan jumlah core untuk instance. Bidang `ThreadsPerCore` menunjukkan jumlah thread per inti.

```
[
    {
        "CoreCount": 24,
        "ThreadsPerCore": 2
    },
]
```

PowerShell

Untuk melihat opsi CPU untuk sebuah instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -InstanceId 'i-1234567890abcdef0').Instances.CpuOptions
```

Berikut ini adalah output contoh.

```
AmdSevSnp CoreCount ThreadsPerCore
```

```
--------- --------- --------------
          24        2
```

Atau, untuk melihat informasi CPU, Anda dapat menyambung ke instance Anda dan menggunakan salah satu alat sistem berikut:

- Windows `Task Manager` pada instans Windows Anda
- Per lscpu intah pada instance Linux Anda

Anda dapat menggunakannya AWS Config untuk merekam, menilai, mengaudit, dan mengevaluasi perubahan konfigurasi untuk instans, termasuk instans yang dihentikan. Untuk informasi selengkapnya, lihat Memulai AWS Config di Panduan Pengguna AWS Config .

## Optimalkan CPU untuk instans License-Included

Beban kerja seperti Microsoft SQL Server sering membutuhkan tingkat memori dan IOPS yang tinggi tetapi jumlah vCPU yang rendah. AWS menyediakan serangkaian jenis instans yang luas yang dapat mencakup sebagian besar kebutuhan infrastruktur Anda. Namun, untuk mengurangi biaya CPU-based lisensi v untuk Windows dan Microsoft SQL Server, Anda dapat menyesuaikan jumlah vCPU yang berjalan pada instans EC2 sambil mempertahankan spesifikasi memori, penyimpanan, dan jaringan yang sama. Pendekatan ini dapat menghemat biaya CPU-based lisensi v untuk beban kerja yang disertakan lisensi dan Bring-Your-Own-License (BYOL). Anda dapat menentukan opsi CPU saat meluncurkan atau memodifikasi instance dengan menggunakan konsol atau Amazon EC2 API. Untuk petunjuk, lihat Tentukan opsi CPU untuk instans Amazon EC2.

Untuk informasi selengkapnya, lihat posting blog ini tentang praktik terbaik untuk mengoptimalkan CPU untuk beban kerja SQL Server.

Jenis lisensi yang didukung

Optimize CPU mendukung penagihan berdasarkan jumlah CPU aktif untuk jenis konfigurasi lisensi berikut untuk instans yang diluncurkan dari AMI yang disertakan lisensi. Untuk informasi selengkapnya tentang jenis lisensi, lihatBidang informasi penagihan AMI.

License-included Penagihan instans AMI

| Lisensi disertakan | Operasi penggunaan | Harga per jam vCPU |
|---|---|---|

| Lisensi disertakan | Operasi penggunaan | Harga per jam vCPU |
|---|---|---|
| Windows Server | RunInstances:0002 | $0.046 |
| Windows Server dengan SQL Server Enterprise | RunInstances:0102 | $0,421 |
| Windows Server dengan Standar SQL Server | RunInstances:0006 | $0.166 |
| Windows Server dengan SQL Server Web | RunInstances:0202 | $0.063 |

Opsi pembelian yang didukung

Optimalkan CPU mendukung opsi pembelian berikut untuk instans yang disertakan dengan lisensi:

- On-Demand
- Paket Tabungan

> **Warning**
>
> Jika Anda menggunakan Instans Cadangan, diskon mungkin tidak diterapkan saat Anda mengonfigurasi Optimalkan CPU untuk instans yang disertakan lisensi di akun pembayar yang sama. Kami menyarankan Anda menggunakan Paket Tabungan untuk mengurangi biaya CPU-based lisensi v dan memberikan penghematan yang sebanding pada biaya komputasi Anda.
>
> Akun yang menggunakan CPU Optimize dan Instans Cadangan untuk Windows dan SQL Server pada jenis instans yang sama sebelum 15 Oktober 2025, ditambahkan ke daftar opt-out untuk mempertahankan pengalaman penagihan mereka saat ini. Untuk memanfaatkan penghematan lisensi Optimalkan CPU, hubungi AWS Dukungan Pusat untuk dihapus dari daftar opt-out.

Cara kerja Optimize CPU untuk menghemat biaya lisensi

Contoh berikut membantu menggambarkan penghematan biaya yang mungkin terjadi saat Anda mengonfigurasi penggunaan CPU Anda.

Contoh 1: Penagihan default Contoh ini menunjukkan instance r7i.8xlarge yang diluncurkan dari Windows dan SQL Server Enterprise AMI yang disertakan lisensi yang berjalan selama 100 jam dengan konfigurasi CPU default 32 vCPU untuk jenis instans (3200 jam vCPU).

Tagihan menunjukkan satu item baris dengan tarif gabungan yang mencakup biaya penggunaan dan lisensi.

| **For illustration purposes only**<br>**Amazon Web Services, Inc. charges by service**<br>**Total services: 1**<br>**Description** | **Usage Quantity** | **Total charges**<br>**$1,558.88**<br>**Amount in USD** |
|---|---|---|
| Elastic Compute Cloud | | $1,558.88 |
| \|- US East (N. Virginia) | | $1,558.88 |
| \|- Amazon Elastic Compute Cloud running Windows and SQL Server Enterprise | | $1,558.88 |
| \| \|- $15.5888 per On Demand Windows with SQL Server Enterprise r7i.8xlarge Instance Hour | 100 Hrs | $1,558.88 |

Contoh 2: Mengoptimalkan penagihan CPU Contoh ini menunjukkan instance r7i.8xlarge yang diluncurkan dari AMI Windows dan SQL Server Enterprise yang disertakan dengan lisensi. Untuk menghemat biaya lisensi, jumlah CPU yang aktif dikurangi menjadi 16 vCPU. Kemudian instance berjalan selama 100 jam dengan konfigurasi baru.

Tagihan menunjukkan dua item baris berikut.

Deskripsi penagihan: Elastic Compute Cloud

Item baris pertama menunjukkan biaya dasar instans Windows dan SQL Server yang berjalan selama 100 jam ($211,68).

Deskripsi penagihan: Amazon EC2 Optimalkan Lisensi CPU Termasuk Biaya Pihak Ketiga

Item baris kedua mencakup biaya lisensi berdasarkan jumlah vCPU yang aktif untuk periode penagihan ($673,60).

| For illustration purposes only<br>Amazon Web Services, Inc. charges by service<br>Total services: 2<br>Description | Usage Quantity | Total charges<br>$885.28<br>Amount in USD |
|---|---|---|
| Elastic Compute Cloud | | $211.68 |
| \|- US East (N. Virginia) | | $211.68 |
| \|- Amazon Elastic Compute Cloud running Windows and SQL Server Enterprise | | $211.68 |
| \| \|- $2.1168 per On Demand Windows and SQL Server Enterprise without licenses r7i.8xlarge Instance Hr | 100 Hrs | $211.68 |
| Amazon EC2 Optimize CPU License Included Third Party Fees | | $673.60 |
| \|- US East (N. Virginia) | | $673.60 |
| \|- Amazon Elastic Compute Cloud running Windows and SQL Server Enterprise | | $673.60 |
| \| \|- $0.421 per vCPU-Hour of On Demand Windows with SQL Server licenses on r7i Instance | 1,600 vCPU-Hrs | $673.60 |

Contoh 3: Mengoptimalkan penagihan CPU dengan Paket Tabungan Contoh ini menunjukkan instance r7i.8xlarge yang diluncurkan dari AMI Windows dan SQL Server Enterprise yang disertakan lisensi. Untuk menghemat biaya lisensi, jumlah CPU yang aktif dikurangi menjadi 16 vCPU. Kemudian instance berjalan selama 100 jam dengan konfigurasi baru.

Rencana Tabungan Tanpa Komputasi Satu tahun dengan $1. 60/hour Komitmen (bulat) memberikan penghematan biaya lebih lanjut yang mengurangi biaya dasar instans Windows dan SQL Server. Komitmen Rencana Tabungan mencakup penggunaan 100 jam penuh dari instance r7i.8xlarge dengan tarif Rencana Tabungan sebesar $1. 53362/hour.

Tagihan menunjukkan item baris berikut.

Deskripsi penagihan: Paket Tabungan untuk penggunaan Komputasi

Item baris pertama menunjukkan komitmen Rencana Tabungan untuk penggunaan 100 jam penuh ($160,00).

Deskripsi penagihan: Elastic Compute Cloud

Item baris kedua berisi dua entri. Entri pertama menunjukkan berapa biaya dasar instans Windows dan SQL Server yang berjalan selama 100 jam tanpa Savings Plan ($211,68). Entri kedua menunjukkan bahwa biaya dasar penuh ditanggung oleh Compute Savings Plan (-$211,68), yang menghasilkan biaya bersih nol untuk item baris ini.

Deskripsi penagihan: Amazon EC2 Optimalkan Lisensi CPU Termasuk Biaya Pihak Ketiga

Item baris ketiga mencakup biaya lisensi berdasarkan jumlah vCPU yang aktif untuk periode penagihan ($673,60).

| For illustration purposes only<br>Amazon Web Services, Inc. charges by service<br>Total services: 3<br>Description | Usage Quantity | Total charges<br>$833.60<br>Amount in USD |
|---|---|---|
| Savings Plans for Compute usage | | $160.00 |
| \|- Any | | $160.00 |
| \|- Compute Savings Plans | | $160.00 |
| \| \|- 1 year No Upfront Compute Savings Plan | 100 Hrs | $160.00 |
| Elastic Compute Cloud | | $0.00 |
| \|- US East (N. Virginia) | | $0.00 |
| \|- Amazon Elastic Compute Cloud running Windows and SQL Server Enterprise | | $0.00 |
| \| \|- $2.1168 per On Demand Windows and SQL Server Enterprise without licenses r7i.8xlarge Instance Hr | 100 Hrs | $211.68 |
| \| \|- r7i.8xlarge Windows with SQL Server Enterprise without licenses instance usage covered by CSP | 100 Hrs | ($211.68) |
| Amazon EC2 Optimize CPU License Included Third Party Fees | | $673.60 |
| \|- US East (N. Virginia) | | $673.60 |
| \|- Amazon Elastic Compute Cloud running Windows and SQL Server Enterprise | | $673.60 |
| \| \|- $0.421 per vCPU-Hour of On Demand Windows with SQL Server licenses on r7i Instance | 1,600 vCPU-Hrs | $673.60 |

# AMD SEV-SNP untuk instans Amazon EC2

AMD Secure En Virtualization-Secure crypted Nested Paging (AMD SEV-SNP) adalah fitur CPU yang menyediakan properti berikut:

- Pengesahan — AMD SEV-SNP memungkinkan Anda mengambil laporan pengesahan yang ditandatangani yang berisi ukuran kriptografi yang dapat digunakan untuk memvalidasi status dan identitas instance, dan bahwa laporan tersebut berjalan pada perangkat keras AMD asli. Untuk informasi selengkapnya, lihat Membuktikan instans Amazon EC2 dengan AMD SEV-SNP.
- Enkripsi memori — Dimulai dengan prosesor AMD EPYC (Milan), AWS Graviton2, dan Intel Xeon Scalable (Ice Lake), memori instance selalu dienkripsi.

Anda dapat menggunakan AMD SEV-SNP dengan Host Khusus atau dengan sewa bersama.

Daftar Isi


- Konsep dan terminologi
- Persyaratan
- Pertimbangan-pertimbangan
- Harga
- Temukan jenis instans EC2 yang mendukung AMD SEV-SNP
- Aktifkan AMD SEV-SNP untuk instans EC2
- Membuktikan instans Amazon EC2 dengan AMD SEV-SNP

## Konsep dan terminologi

Sebelum Anda mulai menggunakan AMD SEV-SNP, pastikan Anda terbiasa dengan konsep dan terminologi berikut.

Laporan peng SEV-SNP esahan AMD

Laporan peng SEV-SNP esahan AMD adalah dokumen yang dapat diminta instance dari CPU. Laporan peng SEV-SNP esahan AMD dapat digunakan untuk memvalidasi status dan identitas suatu instance, dan untuk memverifikasi bahwa itu berjalan di lingkungan AMD yang disetujui. Laporan tersebut mencakup pengukuran peluncuran, yang merupakan hash kriptografi dari status boot awal sebuah instans, termasuk konten memori instans awal dan status awal vCPU. Laporan peng SEV-SNP esahan AMD ditandatangani dengan tanda tangan VCEK (pada Host Khusus) atau tanda tangan VLEK (pada sewa bersama) yang dirantai kembali ke akar kepercayaan AMD.

VCEK

Versioned Chip Endorsement Key (VCEK) adalah kunci penandatanganan per chip yang disertifikasi oleh AMD dan digunakan oleh CPU AMD untuk menandatangani laporan pengesahan AMD pada Host Khus SEV-SNP us. Tanda tangan VCEK dapat divalidasi menggunakan sertifikat yang disediakan oleh AMD.

VLEK

Versioned Loaded Endorsement Key (VLEK) adalah kunci penandatanganan berversi yang disertifikasi oleh AMD dan digunakan oleh CPU AMD untuk menandatangani laporan pengesahan AMD tentang sewa SEV-SNP bersama. Tanda tangan VLEK dapat divalidasi menggunakan sertifikat yang disediakan oleh AMD.

Biner OVMF

Open Virtual Machine Firmware (OVMF) adalah kode boot awal yang digunakan untuk menyediakan lingkungan UEFI untuk instans. Kode boot awal dijalankan sebelum kode di AMI di-boot. OVMF juga menemukan dan menjalankan boot loader yang disediakan di AMI. Untuk informasi selengkapnya, lihat Repositori OVMF.

## Persyaratan

Persyaratan untuk AMD SEV-SNP tergantung pada apakah Anda menggunakan Host Khusus atau sewa bersama.

Persyaratan untuk Host Khusus

Untuk menggunakan AMD SEV-SNP dengan Host Khusus, Anda harus memenuhi persyaratan berikut:

- Gunakan salah satu dari tipe instans yang didukung berikut:
  - Tujuan umum: `m6a.large` `m6a.xlarge` | `m6a.2xlarge` | `m6a.4xlarge` | `m6a.8xlarge`
  - Komputasi yang dioptimalkan: `c6a.large` | `c6a.xlarge` | `c6a.2xlarge` | `c6a.4xlarge` | `c6a.8xlarge` | `c6a.12xlarge` | `c6a.16xlarge`
  - Memori dioptimalkan: `r6a.large` | `r6a.xlarge` | `r6a.2xlarge` | `r6a.4xlarge`
- Alokasikan Host Khusus dengan AMD SEV-SNP diaktifkan. Untuk informasi selengkapnya, lihat Alokasikan Host Khusus dengan AMD SEV-SNP.
- Luncurkan instance Anda di iklan apa pun Wilayah AWS.
- Gunakan AMI dengan `uefi` atau mode `uefi-preferred` boot dan sistem operasi yang mendukung AMD SEV-SNP. Untuk AWS, AMD SEV-SNP didukung pada AL2023, RHEL 9.3, SLES 15 SP4, dan Ubuntu 23.04 dan yang lebih baru.

Persyaratan untuk sewa bersama

Untuk menggunakan AMD SEV-SNP dengan sewa bersama, Anda harus memenuhi persyaratan berikut:

- Gunakan salah satu dari tipe instans yang didukung berikut:
  - Tujuan umum: `m6a.large` `m6a.xlarge` | `m6a.2xlarge` | `m6a.4xlarge` | `m6a.8xlarge`
  - Komputasi yang dioptimalkan: `c6a.large` | `c6a.xlarge` | `c6a.2xlarge` | `c6a.4xlarge` | `c6a.8xlarge` | `c6a.12xlarge` | `c6a.16xlarge`
  - Memori dioptimalkan: `r6a.large` | `r6a.xlarge` | `r6a.2xlarge` | `r6a.4xlarge`
- Luncurkan instans Anda di wilayah yang didukung Wilayah AWS: AS Timur (Ohio) atau Eropa (Irlandia).
- Gunakan AMI dengan `uefi` atau mode `uefi-preferred` boot dan sistem operasi yang mendukung AMD SEV-SNP. Untuk AWS, AMD SEV-SNP didukung pada AL2023, RHEL 9.3, SLES 15 SP4, dan Ubuntu 23.04 dan yang lebih baru.

## Pertimbangan-pertimbangan

Anda hanya dapat mengaktifkan AMD SEV-SNP saat meluncurkan instance. Saat AMD SEV-SNP diaktifkan untuk peluncuran instans Anda, aturan berikut berlaku.

- Setelah diaktifkan, AMD tidak SEV-SNP dapat dinonaktifkan. Itu tetap diaktifkan sepanjang siklus hidup instance.
- Anda hanya dapat mengubah jenis instans ke jenis instans lain yang mendukung AMD SEV-SNP.
- Hibernasi dan Nitro Enclave tidak didukung.

#### Pertimbangan tambahan untuk Host Khusus

Pertimbangan tambahan berikut berlaku saat Anda menggunakan AMD SEV-SNP dengan Host Khusus:

- Pembaruan firmware diterapkan saat Anda mengalokasikan host. Untuk mendapatkan firmware terbaru, lepaskan host yang ada dan alokasikan yang baru. Untuk informasi selengkapnya, lihat Perbarui firmware pada Host SEV-SNP Khusus AMD.
- Anda dapat meluncurkan instans dengan atau tanpa AMD SEV-SNP diaktifkan pada Host Khusus yang memiliki SEV-SNP dukungan AMD.
- Grup Sumber Daya Host dan Reservasi Host Khusus didukung.

#### Pertimbangan tambahan untuk sewa bersama

Pertimbangan tambahan berikut berlaku saat Anda menggunakan AMD SEV-SNP dengan sewa bersama:

- Jika host dasar instans Anda dijadwalkan untuk pemeliharaan, Anda akan menerima pemberitahuan acara terjadwal 14 hari sebelum acara. Anda harus menghentikan atau memulai ulang instans secara manual untuk memindahkannya ke host baru.

## Harga

#### Host Khusus

Tidak ada biaya tambahan untuk menggunakan AMD SEV-SNP di Host Khusus. Anda hanya membayar harga Host Khusus standar. Untuk informasi selengkapnya, lihat Harga Host Khusus.

#### Sewa bersama

Saat meluncurkan instans Amazon EC2 dengan AMD SEV-SNP diaktifkan pada sewa bersama, Anda akan dikenakan biaya penggunaan tambahan per jam yang setara dengan 10 persen dari tarif per jam dari On-Demand jenis instans yang dipilih. Biaya SEV-SNP penggunaan AMD ini merupakan

biaya terpisah untuk penggunaan instans Amazon EC2 Anda. Instans Terpesan, Savings Plans, dan penggunaan sistem operasi tidak memengaruhi biaya ini.

Jika Anda mengonfigurasi Instans Spot untuk diluncurkan dengan AMD SEV-SNP diaktifkan pada sewa bersama, Anda akan dikenakan biaya penggunaan tambahan per jam yang setara dengan 10 persen dari tarif On-Demand per jam dari jenis instans yang dipilih. Jika strategi alokasi menggunakan harga sebagai masukan, Armada Spot tidak termasuk biaya tambahan ini; hanya harga Spot yang digunakan.

## Temukan jenis instans EC2 yang mendukung AMD SEV-SNP

Anda dapat menemukan jenis instance yang mendukung AMD SEV-SNP. Konsol Amazon EC2 tidak menampilkan informasi ini untuk jenis instans.

AWS CLI

Untuk menemukan jenis instans yang mendukung AMD SEV-SNP

Gunakan perintah describe-instance-types berikut ini.

```
aws ec2 describe-instance-types \
    --filters Name=processor-info.supported-features,Values=amd-sev-snp \
    --query 'InstanceTypes[*].[InstanceType]' \
    --output text | sort
```

Berikut ini adalah output contoh.

```
c6a.12xlarge
c6a.16xlarge
c6a.2xlarge
c6a.4xlarge
c6a.8xlarge
c6a.large
c6a.xlarge
m6a.2xlarge
m6a.4xlarge
m6a.8xlarge
m6a.large
m6a.xlarge
r6a.2xlarge
r6a.4xlarge
r6a.large
```

```
r6a.xlarge
```

PowerShell

Untuk menemukan jenis instans yang mendukung AMD SEV-SNP

Gunakan Get-EC2InstanceType cmdlet.

```
(Get-EC2InstanceType `
    -Filter @{Name="processor-info.supported-features"; Values="amd-sev-
snp"}).InstanceType.Value | Sort-Object
```

Berikut ini adalah output contoh.

```
c6a.12xlarge
c6a.16xlarge
c6a.2xlarge
c6a.4xlarge
c6a.8xlarge
c6a.large
c6a.xlarge
m6a.2xlarge
m6a.4xlarge
m6a.8xlarge
m6a.large
m6a.xlarge
r6a.2xlarge
r6a.4xlarge
r6a.large
r6a.xlarge
```

## Aktifkan AMD SEV-SNP untuk instans EC2

Anda dapat meluncurkan instance dengan AMD SEV-SNP diaktifkan. Anda tidak dapat mengaktifkan AMD SEV-SNP setelah peluncuran.

Daftar Isi


- Alokasikan Host Khusus dengan AMD SEV-SNP
- Luncurkan instans di Host SEV-SNP Khusus AMD
- Perbarui firmware pada Host SEV-SNP Khusus AMD

- Luncurkan instance dengan AMD SEV-SNP pada sewa bersama
- Periksa apakah instance EC2 diaktifkan untuk AMD SEV-SNP

### Alokasikan Host Khusus dengan AMD SEV-SNP

Sebelum Anda dapat meluncurkan instans dengan AMD SEV-SNP di Host Khusus, Anda harus mengalokasikan Host Khusus dengan SEV-SNP dukungan AMD diaktifkan. Anda mengalokasikan Host Khusus berdasarkan keluarga instans (misalnya, m6a, r6a, atau c6a), dan kemudian Anda dapat menjalankan jenis instans yang didukung dari keluarga tersebut di host. Anda dapat mengalokasikan Host Khusus dengan AMD SEV-SNP dalam iklan apa pun Wilayah AWS.

#### Console

Untuk mengalokasikan Host Khusus dengan AMD SEV-SNP diaktifkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih Alo kasikan Host Khusus.
4. Untuk AMD SEV-SNP, pilih Aktifkan.
5. Konfigurasikan setelan host yang tersisa sesuai kebutuhan, lalu pilih Alo kasikan.

#### AWS CLI

Untuk mengalokasikan Host Khusus dengan AMD SEV-SNP diaktifkan

Gunakan allocate-hosts perintah dengan `--cpu-options` parameter `--instance-type` dan.

```
aws ec2 allocate-hosts \
    --instance-type "m6a.large" \
    --availability-zone "us-east-1a" \
    --auto-placement "off" \
    --cpu-options AmdSevSnp=enabled \
    --quantity 1
```

Berikut ini adalah output contoh.

```
{
    "HostIds": [
        "h-0123456789abcdef0"
```

```
    ]
}
```

> **Note**
>
> Ketika Anda mengalokasikan Host Khusus dengan AMD SEV-SNP diaktifkan, host memasuki `configuring` status saat AWS menerapkan firmware terbaru. Ini bisa memakan waktu dari beberapa menit hingga 2 jam. Anda dikenakan biaya selama `configuring` negara.

### Luncurkan instans di Host SEV-SNP Khusus AMD

Anda dapat meluncurkan instance dengan AMD SEV-SNP diaktifkan setelah Host Khusus dikonfigurasi.

> **Note**
>
> Anda juga dapat meluncurkan instans tanpa AMD SEV-SNP diaktifkan pada Host Khusus yang dialokasikan dengan AMD SEV-SNP. SEV-SNP Instans AMD SEV-SNP dan non-AMD dapat berjalan berdampingan satu sama lain pada host yang sama.

#### AWS CLI

Untuk meluncurkan instans dengan AMD SEV-SNP pada Host Khusus

Gunakan run-instances perintah dengan `--placement` opsi `--cpu-options` dan. Ganti *`h-0123456789abcdef0`* dengan ID Host Khusus yang dialokasikan.

```
aws ec2 run-instances \
    --instance-type m6a.xlarge \
    --image-id ami-0123456789example \
    --cpu-options AmdSevSnp=enabled \
    --placement HostId=h-0123456789abcdef0
```

#### PowerShell

Untuk meluncurkan instans dengan AMD SEV-SNP pada Host Khusus

Gunakan New-EC2Instance cmdlet dengan parameter `-CpuOption` dan`-Placement_HostId`.

```
New-EC2Instance `
    -InstanceType m6a.xlarge `
    -ImageId ami-0123456789example `
    -CpuOption @{AmdSevSnp="enabled"} `
    -Placement_HostId h-0123456789abcdef0
```

Perbarui firmware pada Host SEV-SNP Khusus AMD

Saat Anda mengalokasikan Host Khusus dengan AMD SEV-SNP diaktifkan AWS , terapkan firmware terbaru yang tersedia ke host. Pembaruan firmware dapat mencakup patch keamanan dan peningkatan fitur dari AMD.

Untuk menerapkan pembaruan firmware, alokasikan host baru (yang menerima firmware terbaru), pindahkan instance Anda ke host baru, lalu lepaskan host lama.

1. Alokasikan Host Khusus baru dengan AMD SEV-SNP diaktifkan. Host baru menerima firmware terbaru yang tersedia. Untuk informasi selengkapnya, lihat Alokasikan Host Khusus dengan AMD SEV-SNP.
2. Hentikan instans yang sedang berjalan di Host Khusus lama.

```
aws ec2 stop-instances \
    --instance-ids i-0123456789example
```

3. Ubah penempatan instans untuk menargetkan Host Khusus yang baru. Gunakan perintah modify-instance-placement.

```
aws ec2 modify-instance-placement \
    --instance-id i-0123456789example \
    --host-id h-09876543210abcdef
```

4. Mulai instans di Host Khusus baru.

```
aws ec2 start-instances \
    --instance-ids i-0123456789example
```

5. Lepaskan Host Dedicated lama. Gunakan perintah release-hosts.

```
aws ec2 release-hosts \
    --host-ids h-0123456789abcdef0
```

> **Note**
>
> Anda menerima pemberitahuan pemeliharaan untuk Host Khusus Anda dengan cara yang sama seperti Anda menerimanya untuk acara pemeliharaan Host Khusus lainnya. Saat Anda menerima pemberitahuan, ikuti langkah-langkah sebelumnya untuk pindah ke host baru dengan firmware yang diperbarui.

Luncurkan instance dengan AMD SEV-SNP pada sewa bersama

Anda dapat meluncurkan instance dengan AMD SEV-SNP pada sewa bersama di AS Timur (Ohio) dan Eropa (Irlandia).

AWS CLI

Untuk meluncurkan instance dengan AMD SEV-SNP pada sewa bersama

Gunakan perintah run-instances dengan opsi `--cpu-options`.

```
aws ec2 run-instances \
    --instance-type m6a.xlarge \
    --image-id ami-0123456789example \
    --cpu-options AmdSevSnp=enabled
```

PowerShell

Untuk meluncurkan instance dengan AMD SEV-SNP pada sewa bersama

Gunakan New-EC2Instance cmdlet dengan parameter. `-CpuOption`

```
New-EC2Instance `
    -InstanceType m6a.xlarge `
    -ImageId ami-0123456789example `
    -CpuOption @{AmdSevSnp="enabled"}
```

Periksa apakah instance EC2 diaktifkan untuk AMD SEV-SNP

Anda dapat menemukan instance yang diaktifkan untuk AMD SEV-SNP. Konsol Amazon EC2 tidak menampilkan informasi ini.

AWS CLI

Untuk memeriksa apakah AMD SEV-SNP diaktifkan untuk sebuah instance

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[].Instances[].CpuOptions
```

Berikut ini adalah output contoh. Jika tidak `AmdSevSnp` ada di`CpuOptions`, maka AMD dinon SEV-SNP aktifkan.

```
[
    {
        "AmdSevSnp": "enabled",
        "CoreCount": 1,
        "ThreadsPerCore": 2
    }
]
```

PowerShell

Untuk memeriksa apakah AMD SEV-SNP diaktifkan untuk sebuah instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances.CpuOptions
```

Berikut ini adalah output contoh. Jika nilainya tidak `AmdSevSnp` ada, maka AMD dinon SEV-SNP aktifkan.

```
AmdSevSnp CoreCount ThreadsPerCore
--------- --------- --------------
enabled   1         2
```

AWS CloudTrail

Dalam hal AWS CloudTrail permintaan peluncuran instans, properti berikut menunjukkan bahwa AMD SEV-SNP diaktifkan untuk instance.

```
"cpuOptions": {"AmdSevSnp": "enabled"}
```

## Membuktikan instans Amazon EC2 dengan AMD SEV-SNP

Pengesahan adalah proses yang memungkinkan instans Anda membuktikan status dan identitasnya. Setelah mengaktifkan AMD SEV-SNP untuk instance Anda, Anda dapat meminta laporan pengesahan AMD SEV-SNP dari prosesor yang mendasarinya. Laporan peng SEV-SNP esahan AMD berisi hash kriptografi, yang disebut pengukuran peluncuran, dari konten memori tamu awal dan status vCPU awal. Laporan pengesahan ditandatangani dengan tanda tangan VCEK (pada Host Khusus) atau tanda tangan VLEK (pada sewa bersama) yang dirantai kembali ke akar kepercayaan AMD. Anda dapat menggunakan pengukuran peluncuran yang disertakan dalam laporan pengesahan untuk memvalidasi bahwa instans berjalan di lingkungan AMD asli dan untuk memvalidasi kode boot awal yang digunakan untuk meluncurkan instans.

### Prasyarat

Luncurkan instance yang diaktifkan untuk AMD SEV-SNP. Untuk informasi selengkapnya, lihat Aktifkan AMD SEV-SNP untuk instans EC2.

### Daftar Isi


- Pengesahan untuk instans pada Host Khusus
- Pengesahan untuk contoh tentang sewa bersama


### Pengesahan untuk instans pada Host Khusus

Host Khusus menggunakan Versioned Chip Endorsement Key (VCEK), kunci penandatanganan per chip yang unik, untuk menandatangani laporan pengesahan. Anda harus memvalidasi rantai kepercayaan dari sertifikat VCEK kembali ke akar kepercayaan AMD.

> **Note**
>
> Model prosesor yang didukung saat ini adalah`milan`.

#### Prasyarat

Instans yang berjalan pada Host Khusus yang di SEV-SNP aktifkan AMD dengan Git, Cargo, dan Perl diinstal.

Langkah 1: Bangun utilitas snpguest

Pada langkah ini, Anda menginstal dan membangun `snpguest` utilitas, yang Anda gunakan untuk menghasilkan laporan pengesahan, mengambil sertifikat yang diperlukan, dan memvalidasi laporan pengesahan.

1. Terhubung ke instans Anda.
2. Verifikasi prasyarat diinstal.

```
$ perl --version; cargo --version; git --version
```

3. Jalankan perintah berikut untuk membangun `snpguest` utilitas dari snpguest repository.

```
$ git clone https://github.com/virtee/snpguest.git
$ cd snpguest
$ cargo build -r
$ cd target/release
```

Langkah 2: Buat laporan pengesahan

Hasilkan permintaan untuk laporan pengesahan. `snpguest`Utilitas meminta laporan pengesahan dari AMD Secure Processor melalui host, dan menulisnya ke file biner. Contoh berikut membuat permintaan acak nonce dan menyimpan laporan di`report.bin`.

```
$ ./snpguest report report.bin request-file.txt --random
```

Langkah 3: Ambil dan validasi rantai sertifikat VCEK

Laporan pengesahan ditandatangani oleh VCEK, yang unik untuk chip AMD di Host Khusus Anda. Anda harus memvalidasi rantai kepercayaan dari VCEK kembali ke akar kepercayaan AMD.

1. Ambil sertifikat VCEK dari Layanan Distribusi Kunci AMD (KDS). Sertifikat diidentifikasi oleh ID chip dan versi TCB dari laporan pengesahan.

```
$ ./snpguest fetch vcek pem ./ ./report.bin --processor-model milan
```

2. Ambil sertifikat root AMD (AMD Root Key (ARK) dan AMD SEV Key (ASK)) yang membentuk rantai kepercayaan.

```
$ ./snpguest fetch ca PEM ./ milan --endorser vcek
```

3. Verifikasi rantai sertifikat.

```
$ ./snpguest verify certs ./
```

Berikut ini adalah output contoh.

```
The AMD ARK was self-signed!
The AMD ASK was signed by the AMD ARK!
The VCEK was signed by the AMD ASK!
```

Opsional: Untuk transparansi tambahan, Anda dapat memverifikasi rantai secara independen dengan mengunduh sertifikat langsung dari AMD dan menggunakan`openssl`.

```
$ curl --proto '=https' --tlsv1.2 -sSf https://kdsintf.amd.com/vcek/v1/Milan/cert_chain
 -o ./cert_chain.pem
$ openssl verify --CAfile ./cert_chain.pem vcek.pem
```

Berikut ini adalah output contoh.

```
vcek.pem: OK
```

Langkah 4: Validasi tanda tangan laporan pengesahan

Verifikasi bahwa laporan pengesahan ditandatangani oleh sertifikat VCEK. Ini menegaskan bahwa laporan dibuat pada perangkat keras AMD asli dan belum dirusak.

```
$ ./snpguest verify attestation ./ report.bin
```

Berikut ini adalah output contoh.

```
Reported TCB Boot Loader from certificate matches the attestation report.
Reported TCB TEE from certificate matches the attestation report.
Reported TCB SNP from certificate matches the attestation report.
Reported TCB Microcode from certificate matches the attestation report.
VEK signed the Attestation Report!
```

Bidang versi TCB (Trusted Computing Base) mengonfirmasi bahwa versi firmware dalam sertifikat cocok dengan yang dilaporkan dalam laporan pengesahan. Baris terakhir menegaskan VCEK menandatangani laporan tersebut.

Pengesahan untuk contoh tentang sewa bersama

Instans sewa bersama menggunakan Versioned Loaded Endorsement Key (VLEK), yang dikeluarkan oleh AMD untuk. AWS Anda harus memvalidasi rantai kepercayaan dari sertifikat VLEK kembali ke akar kepercayaan AMD.

Langkah 1: Bangun utilitas snpguest

Pada langkah ini, Anda menginstal dan membangun `snpguest` utilitas, yang Anda gunakan untuk menghasilkan laporan pengesahan, mengambil sertifikat yang diperlukan, dan memvalidasi laporan pengesahan.

1. Terhubung ke instans Anda.
2. Jalankan perintah berikut untuk membangun `snpguest` utilitas dari snpguest repository.

```
$ git clone https://github.com/virtee/snpguest.git
$ cd snpguest
$ cargo build -r
$ cd target/release
```

Langkah 2: Hasilkan laporan pengesahan

Hasilkan permintaan untuk laporan pengesahan. `snpguest`Utilitas meminta laporan pengesahan dari host, dan menulisnya ke file biner. Contoh berikut membuat permintaan acak nonce dan menyimpan laporan di`report.bin`.

```
$ ./snpguest report report.bin request-file.txt --random
```

Minta sertifikat dari memori host, dan simpan sebagai file PEM.

```
$ ./snpguest certificates PEM ./
```

Langkah 3: Ambil dan validasi rantai sertifikat VLEK

Laporan pengesahan ditandatangani oleh VLEK, yang dikeluarkan oleh AMD untuk. AWS Anda harus memvalidasi rantai kepercayaan dari VLEK kembali ke akar kepercayaan AMD.

1. Unduh root sertifikat kepercayaan VLEK dari Layanan Distribusi Kunci AMD resmi ke direktori saat ini.

```
$ sudo curl --proto '=https' --tlsv1.2 -sSf https://kdsintf.amd.com/vlek/v1/Milan/
cert_chain -o ./cert_chain.pem
```

2. Gunakan `openssl` untuk memvalidasi bahwa sertifikat VLEK ditandatangani oleh root sertifikat kepercayaan AMD.

```
$ sudo openssl verify --CAfile ./cert_chain.pem vlek.pem
```

   Berikut ini adalah output contoh.

```
vlek.pem: OK
```

Langkah 4: Validasi tanda tangan laporan pengesahan

Verifikasi bahwa laporan pengesahan ditandatangani oleh sertifikat VLEK. Ini menegaskan bahwa laporan dibuat pada perangkat keras AMD asli dan belum dirusak.

```
$ ./snpguest verify attestation ./ report.bin
```

Berikut ini adalah output contoh.

```
Reported TCB Boot Loader from certificate matches the attestation report.
Reported TCB TEE from certificate matches the attestation report.
Reported TCB SNP from certificate matches the attestation report.
Reported TCB Microcode from certificate matches the attestation report.
VEK signed the Attestation Report!
```

## Kontrol status prosesor untuk instans Amazon EC2 Linux

C-statesmengontrol tingkat tidur yang dapat dimasukkan inti saat menganggur. C-states diberi nomor dimulai dengan C0 (keadaan terdangkal di mana inti benar-benar terjaga dan menjalankan instruksi) dan pergi ke C6 (keadaan idle terdalam di mana inti dimatikan).

P-statesmengontrol kinerja yang diinginkan (dalam frekuensi CPU) dari inti. P-states diberi nomor mulai dari P0 (pengaturan kinerja tertinggi di mana inti diizinkan untuk menggunakan Teknologi Intel

Turbo Boost untuk meningkatkan frekuensi jika memungkinkan), dan mereka beralih dari P1 ( P-state yang meminta frekuensi dasar maksimum) ke P15 (frekuensi serendah mungkin).

> **Note**
>
> AWS Prosesor Graviton memiliki mode hemat daya bawaan dan beroperasi pada frekuensi tetap. Oleh karena itu, mereka tidak memberikan kemampuan bagi sistem operasi untuk mengontrol C-states dan P-states.

### C-states dan P-states

Jenis instance berikut menyediakan kemampuan bagi sistem operasi untuk mengontrol C-states dan P-states:

- Tujuan umum: `m4.10xlarge` | `m4.16xlarge`
- Komputasi dioptimalkan: `c4.8xlarge`
- Memori dioptimalkan: `r4.8xlarge` `r4.16xlarge` `x1.16xlarge` | `x1.32xlarge` | `x1e.8xlarge` | `x1e.16xlarge` | `x1e.32xlarge`
- Penyimpanan yang dioptimalkan: `d2.8xlarge` | `i3.8xlarge` | `i3.16xlarge` | `h1.8xlarge` | `h1.16xlarge`
- Komputasi yang dipercepat: `g3.16xlarge` | `p3.16xlarge`
- Bare metal: Semua instance bare metal dengan prosesor Intel dan AMD

### C-states hanya

Jenis instance berikut menyediakan kemampuan sistem operasi untuk mengontrol C-states:

- Tujuan umum: `m5.12xlarge` `m5.24xlarge` | `m5d.12xlarge` | `m5d.24xlarge` | `m5n.12xlarge` | `m5n.24xlarge` | `m5dn.12xlarge` | `m5dn.24xlarge` | `m5zn.6xlarge` | `m5zn.12xlarge` | `m6a.24xlarge` `m6a.48xlarge` | `m6i.16xlarge` | `m6i.32xlarge` | `m6id.16xlarge` | `m6id.32xlarge` | `m6idn.16xlarge` | `m6in.16xlarge` | `m6in.32xlarge` | `m7a.medium` `m7a.large` | `m7a.xlarge` | `m7a.2xlarge` | `m7a.4xlarge` | `m7a.8xlarge` | `m7a.12xlarge` | `m7a.16xlarge` | `m7a.24xlarge` | `m7a.32xlarge` | `m7a.48xlarge` `m7i.large` | `m7i.xlarge` | `m7i.2xlarge` | `m7i.4xlarge` | `m7i.8xlarge` | `m7i.12xlarge` | | | `m7i.16xlarge` | | `m7i.24xlarge` | | `m7i.48xlarge` | | `m8a.medium` | | `m8a.large` | | `m8a.xlarge` | | `m8a.2xlarge` | | `m8a.4xlarge` | | `m8a.8xlarge` | | `m8a.12xlarge` | |

m8a.16xlarge||m8a.24xlarge|||m8a.48xlarge||m8azn.medium||m8azn.large
||m8azn.xlarge||m8azn.3xlarge||m8azn.6xlarge||m8azn.12xlarge||
m8azn.24xlarge||m8i.large||m8i.xlarge||m8i.2xlarge||m8i.4xlarge|||
m8i.8xlarge||m8i.12xlarge||m8i.16xlarge||m8i.24xlarge||m8i.32xlarge|
|m8i.48xlarge||m8i.96xlarge||m8id.large||m8id.xlarge||m8id.2xlarge||
m8id.4xlarge||m8id.8xlarge||m8id.12xlarge||m8id.16xlarge|||m8id.24xlarge
||m8id.32xlarge||m8id.48xlarge||m8id.96xlarge||m8in.large||m8in.xlarge|
|m8in.2xlarge||m8in.4xlarge||m8in.8xlarge||m8in.12xlarge||m8in.16xlarge
||m8in.24xlarge|||m8in.32xlarge||m8in.48xlarge||m8in.96xlarge||
m8idn.large||m8idn.xlarge||m8idn.2xlarge||m8idn.4xlarge||m8idn.8xlarge
||m8idn.12xlarge||m8idn.16xlarge||m8idn.24xlarge||m8idn.32xlarge|||
m8idn.48xlarge||m8idn.96xlarge||m8ib.large||m8ib.xlarge||m8ib.2xlarge||
m8ib.4xlarge||m8ib.8xlarge||m8ib.12xlarge||m8ib.16xlarge||m8ib.24xlarge
||m8ib.32xlarge||m8ib.48xlarge||m8ib.96xlarge||m8idb.large| m8idb.xlarge |
m8idb.2xlarge | m8idb.4xlarge | m8idb.8xlarge | m8idb.12xlarge | m8idb.16xlarge |
m8idb.24xlarge | m8idb.32xlarge | m8idb.48xlarge | m8idb.96xlarge | m8ine.large |
m8ine.xlarge | m8ine.2xlarge | m8ine.4xlarge | m8ine.8xlarge | m8ine.12xlarge

- Komput asi dioptimalkan: c5.9xlarge c5.12xlarge c5.18xlarge c5.24xlarge
  c5a.24xlarge | c5ad.24xlarge | c5d.9xlarge | c5d.12xlarge | c5d.18xlarge
  c5d.24xlarge | c5n.9xlarge | c5n.18xlarge | c6a.24xlarge | c6a.32xlarge
  c6a.48xlarge | c6i.16xlarge | c6i.32xlarge | c6id.24xlarge | c6id.32xlarge |
  c6in.32xlarge c7a.medium | c7a.large | c7a.xlarge | c7a.2xlarge | c7a.4xlarge
  c7a.8xlarge | c7a.12xlarge | c7a.16xlarge | c7a.24xlarge | c7a.32xlarge |
  c7a.48xlarge c7i.large | c7i.xlarge | c7i.2xlarge | c7i.4xlarge c7i.8xlarge|
  ||c7i.12xlarge||c7i.16xlarge||c7i.24xlarge||c7i.48xlarge||c8a.medium
  ||c8a.large||c8a.xlarge||c8a.2xlarge||c8a.4xlarge||c8a.8xlarge||
  c8a.12xlarge||c8a.16xlarge|||c8a.24xlarge||c8a.48xlarge||c8i.large
  ||c8i.xlarge||c8i.2xlarge||c8i.4xlarge||c8i.8xlarge||c8i.12xlarge||
  c8i.16xlarge||c8i.24xlarge||c8i.32xlarge||c8i.48xlarge|||c8i.96xlarge
  ||c8id.large||c8id.xlarge||c8id.2xlarge||c8id.4xlarge||c8id.8xlarge
  ||c8id.12xlarge||c8id.16xlarge||c8id.24xlarge||c8id.32xlarge||
  c8id.48xlarge||c8id.96xlarge||c8in.large||c8in.xlarge|||c8in.2xlarge||
  c8in.4xlarge||c8in.8xlarge||c8in.12xlarge||c8in.16xlarge||c8in.24xlarge
  ||c8in.32xlarge||c8in.48xlarge||c8in.96xlarge||c8ib.large||c8ib.xlarge||
  ||c8ib.2xlarge||c8ib.4xlarge||c8ib.8xlarge||c8ib.12xlarge||c8ib.16xlarge|
  ||c8ib.24xlarge||c8ib.32xlarge||c8ib.48xlarge||c8ib.96xlarge||c8in.large

|| c8in.xlarge || c8in.2xlarge || c8in.4xlarge ||| c8in.8xlarge || c8in.12xlarge || x8i.large || x8i.xlarge || x8i.2xlarge || x8i.4xlarge || x8i.8xlarge || x8i.12xlarge || x8i.16xlarge || x8i.24xlarge || x8i.32xlarge || x8i.48xlarge || x8i.48xlarge| x8i.96xlarge

- Memori dioptimalkan: r5.12xlarge r5.24xlarge r5b.12xlarge r5d.12xlarge | r5d.24xlarge | r5n.12xlarge r5n.24xlarge | r5dn.12xlarge | r5dn.24xlarge | r6a.24xlarge | r6a.48xlarge | r6i.16xlarge r6i.32xlarge | r6id.16xlarge | r6id.32xlarge | r6in.16xlarge | r6in.32xlarge | r7a.medium | r7a.large r7a.xlarge | r7a.2xlarge | r7a.4xlarge | r7a.8xlarge | r7a.12xlarge | r7a.16xlarge r7a.24xlarge | r7a.32xlarge | r7a.48xlarge | r7i.large | r7i.xlarge | r7i.2xlarge r7i.4xlarge | r7i.8xlarge | r7i.12xlarge | r7i.16xlarge | r7i.24xlarge ||| r7i.48xlarge || r7iz.large || r7iz.xlarge || r7iz.2xlarge || r7iz.4xlarge || r7iz.8xlarge || r7iz.12xlarge || r7iz.16xlarge || r7iz.32xlarge || r8a.medium || r8a.large || r8a.xlarge ||| r8a.2xlarge || r8a.4xlarge || r8a.8xlarge || r8a.12xlarge || r8a.16xlarge || r8a.24xlarge || r8a.48xlarge || r8i.large || r8i.xlarge || r8i.2xlarge || r8i.4xlarge || r8i.8xlarge ||| r8i.12xlarge || r8i.16xlarge || r8i.24xlarge || r8i.32xlarge || r8i.48xlarge || r8i.96xlarge || r8id.large || r8id.xlarge || r8id.2xlarge || r8id.4xlarge || r8id.8xlarge || r8id.12xlarge || r8id.16xlarge || r8id.24xlarge ||| r8id.32xlarge || r8id.48xlarge || r8id.96xlarge || r8in.large || r8in.xlarge || r8in.2xlarge || r8in.4xlarge || r8in.8xlarge || r8in.12xlarge || r8in.16xlarge || r8in.24xlarge || r8in.32xlarge ||| r8in.48xlarge || r8in.96xlarge || r8idn.large || r8idn.xlarge || r8idn.2xlarge || r8idn.4xlarge || r8idn.8xlarge || r8idn.12xlarge || r8idn.16xlarge || r8idn.24xlarge || r8idn.32xlarge || r8idn.48xlarge ||| r8idn.96xlarge || r8ib.large || r8ib.xlarge || r8ib.2xlarge || r8ib.4xlarge || r8ib.8xlarge || r8ib.12xlarge || r8ib.16xlarge || r8ib.24xlarge || r8ib.32xlarge || r8ib.48xlarge || r8ib.96xlarge || r8idb.large || r8idb.xlarge||| r8idb.2xlarge || r8idb.4xlarge || r8idb.8xlarge || r8idb.12xlarge || r8idb.16xlarge || r8idb.24xlarge || r8idb.32xlarge || r8idb.48xlarge || r8idb.96xlarge || u-3tb1.56xlarge || u-6tb1.56xlarge ||| u-6tb1.112xlarge || u-9tb1.112xlarge || u-12tb1.112xlarge || u-18tb1.112xlarge || u-24tb1.112xlarge || u7i-6tb.112xlarge || u7i-8tb.112xlarge || u7i-12tb.224xlarge || u7in-16tb.224xlarge || u7in-24tb.224xlarge | | u7in-32tb.224xlarge ||| u7inh-32tb.480xlarge || x2idn.32xlarge || x2iedn.16xlarge || x2iedn.32xlarge || x2iezn.12xlarge || x8aedz.large |

| x8aedz.xlarge || x8aedz.3xlarge || x8aedz.6xlarge || x8aedz.12xlarge || x8aedz.24xlarge || z1d.6xlarge || z1d.12xlarge

- Penyimpanan dioptimalkan: d3en.12xlarge dl1.24xlarge i3en.12xlarge i3en.24xlarge | i4i.16xlarge | i7i.large | i7i.xlarge i7i.2xlarge | i7i.4xlarge | i7i.8xlarge | i7i.12xlarge | i7i.16xlarge | i7i.24xlarge i7i.48xlarge | i7ie.large | i7ie.xlarge | i7ie.2xlarge | i7ie.3xlarge | i7ie.6xlarge | i7ie.12xlarge i7ie.18xlarge | i7ie.24xlarge | i7ie.48xlarge | r5b.12xlarge | r5b.24xlarge
- Komputasi yang dipercepat: dl1.24xlarge f2.6xlarge f2.12xlarge f2.48xlarge g5.24xlarge g5.48xlarge | g6.24xlarge | g6.48xlarge | g6e.12xlarge | g6e.24xlarge | g6e.48xlarge g7e.2xlarge | g7e.4xlarge | g7e.8xlarge | g7e.12xlarge | g7e.24xlarge | g7e.48xlarge | g7.2xlarge | g7.4xlarge | g7.8xlarge g7.12xlarge | g7.24xlarge | g7.48xlarge | inf1.24xlarge | p3dn.24xlarge | p4d.24xlarge | p4de.24xlarge | p5.48xlarge | p5e.48xlarge | p5en.48xlarge p6-b200.48xlarge | p6-b300.48xlarge | trn1.32xlarge | trn2.3xlarge | trn2.48xlarge | trn2a.3xlarge| trn2a.48xlarge trn2n.3xlarge | trn2n.48xlarge | trn2p.48xlarge | trn2u.48xlarge | vt1.24xlarge
- High-performance komputasi: hpc7a.12xlarge hpc7a.24xlarge || hpc7a.48xlarge | hpc7a.96xlarge | hpc8a.96xlarge

Anda mungkin ingin mengubah P-state pengaturan C-state atau untuk meningkatkan konsistensi kinerja prosesor, mengurangi latensi, atau menyesuaikan instance Anda untuk beban kerja tertentu. Default C-state dan P-state pengaturan memberikan kinerja maksimum, yang optimal untuk sebagian besar beban kerja. Namun, jika aplikasi Anda akan mendapat manfaat dari pengurangan latensi dengan mengorbankan frekuensi single atau dual-core yang lebih tinggi, atau dari kinerja yang konsisten pada frekuensi yang lebih rendah dibandingkan frekuensi Turbo Boost yang meledak, pertimbangkan untuk bereksperimen dengan P-state pengaturan C-state atau yang tersedia untuk instans ini.

Untuk informasi tentang konfigurasi prosesor yang berbeda dan cara memantau efek konfigurasi Anda untuk Amazon Linux, lihat Kontrol status prosesor untuk instans Amazon EC2 Amazon Linux di Panduan Pengguna Amazon Linux 2. Prosedur ini ditulis untuk, dan berlaku untuk Amazon Linux; Namun, mereka mungkin juga bekerja untuk distribusi Linux lain dengan kernel Linux 3.9 atau lebih baru. Untuk informasi lebih lanjut tentang distribusi Linux dan kontrol status prosesor lainnya, lihat dokumentasi khusus sistem Anda.

# Instans terkelola Amazon EC2

Instans terkelola  Amazon EC2  adalah instans EC2 yang disediakan dan dikelola oleh penyedia layanan yang ditunjuk. Instans terkelola menyediakan cara yang disederhanakan untuk menjalankan beban kerja komputasi di Amazon EC2 dengan memungkinkan Anda mendelegasikan kontrol operasional instans ke penyedia layanan.

Kontrol yang didelegasikan adalah satu-satunya perubahan yang diperkenalkan untuk instans terkelola. Spesifikasi teknis dan penagihan tetap sama dengan instans EC2 yang tidak dikelola. Karena instans terkelola memungkinkan Anda mendelegasikan kontrol ke penyedia layanan, Anda dapat memanfaatkan keahlian operasional dan praktik terbaik penyedia layanan. Ketika instance dikelola, penyedia layanan bertanggung jawab atas tugas-tugas seperti menyediakan instance, mengonfigurasi perangkat lunak, kapasitas penskalaan, menangani kegagalan dan penggantian instans, dan menghentikan instance.

Anda tidak dapat secara langsung mengubah pengaturan instance terkelola atau menghentikannya. Layanan dan operasi spesifik ditentukan oleh perjanjian antara Anda dan penyedia layanan. Namun, Anda dapat menambahkan, memodifikasi, atau menghapus tag dari instans terkelola, memungkinkan Anda untuk mengkategorikannya dalam AWS lingkungan Anda.

AWS Layanan berikut menawarkan instans terkelola:

- Mode Otomatis EKS
- Instans terkelola Amazon ECS
- WorkSpaces Instans terkelola inti
- Instans terkelola Lambda
- Peluncuran Cepat Amazon EC2
- Instans AgentCore runtime batuan

Daftar Isi


- Penagihan untuk instans terkelola
- Mengidentifikasi instans terkelola
- Pengaturan visibilitas sumber daya terkelola
- Memulai dengan instans terkelola

## Penagihan untuk instans terkelola

Instans terkelola Amazon EC2 dikenakan biaya dasar yang sama dengan instans Amazon EC2 yang tidak dikelola, ditambah biaya terpisah untuk penyedia layanan. Biaya tambahan ini dibebankan oleh penyedia layanan yang mengelola instans Anda dan ditagih secara terpisah. Ini mencakup biaya layanan yang disediakan untuk mengoperasikan dan memelihara instans terkelola Anda.

Semua opsi pembelian Amazon EC2 tersedia untuk instans terkelola, termasuk Instans, On-Demand Instans Cadangan, Instans Spot, dan Paket Tabungan. Dengan mencari sumber komputasi Anda langsung dari EC2 dan kemudian memberikannya kepada penyedia layanan Anda, Anda mendapatkan keuntungan dari Instans Cadangan atau Paket Tabungan yang ada yang diterapkan ke akun Anda, memastikan bahwa Anda menggunakan kapasitas komputasi paling hemat biaya yang tersedia.

Misalnya, saat menggunakan Mode Otomatis Amazon EKS, Anda membayar tarif instans EC2 standar untuk instans yang mendasarinya, ditambah biaya tambahan dari Amazon EKS untuk mengelola instans atas nama Anda. Jika Anda kemudian memutuskan untuk mendaftar ke Paket Tab ungan, tarif instans EC2 dikurangi oleh Paket Tabungan, sementara biaya tambahan dari Amazon EKS tetap tidak berubah.

## Mengidentifikasi instans terkelola

Instans terkelola diidentifikasi dengan nilai sebenarnya di bidang Terkelola. Penyedia layanan diidentifikasi di bidang Operator (di konsol) atau `Principal` bidang (di CLI).

Gunakan prosedur berikut untuk mengidentifikasi instans terkelola.

Console

Untuk mengidentifikasi instance terkelola

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance yang ingin Anda periksa.
4. Pada tab Detail (jika Anda memilih kotak centang) atau di area ringkasan (jika Anda memilih ID instance), cari bidang Terkel ola.
   - Nilai true menunjukkan instance terkelola.
   - Nilai false menunjukkan instance yang tidak dikelola.

5. Jika Ter kelola disetel ke true, bidang Operator menampilkan nilai yang mengidentifikasi penyedia layanan yang bertanggung jawab untuk mengelola instance. Misalnya, nilai eks.amazonaws.com mengidentifikasi Amazon EKS sebagai penyedia layanan.

AWS CLI

Untuk mengidentifikasi instance terkelola

Gunakan perintah describe-instances dan tentukan ID instans.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[].Instances[].Operator
```

Berikut ini adalah output contoh. Jika `Managed` ya`true`, instance tersebut adalah instance terkelola dan `Principal` disertakan. Prinsipal adalah penyedia layanan yang mengelola instance. Misalnya, nilai `eks.amazonaws.com` mengidentifikasi Amazon EKS sebagai penyedia layanan.

```
[
    {
        "Managed": true,
        "Principal": "eks.amazonaws.com"
    }
]
```

Untuk menemukan instans terkelola

Gunakan perintah describe-instance dan tentukan `operator.managed` filter dengan nilai. `true` `--query`Opsi ini hanya menampilkan ID dari instance terkelola.

```
aws ec2 describe-instances \
    --filters "Name=operator.managed,Values=true" \
    --query Reservations[*].Instances[].InstanceId
```

PowerShell

Untuk mengidentifikasi instance terkelola

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances.Operator
```

Berikut ini adalah output contoh.

```
Managed Principal
------- ---------
True    eks.amazonaws.com
```

Untuk menemukan instans terkelola

Gunakan Get-EC2Instance cmdlet. Contoh ini hanya menampilkan ID dari instance terkelola.

```
(Get-EC2Instance -Filter @{Name="operator.managed";
 Values="true"}).Instances.InstanceId
```

## Pengaturan visibilitas sumber daya terkelola

Anda dapat mengontrol apakah sumber daya yang AWS disediakan layanan atas nama Anda akan muncul di tampilan konsol Amazon EC2 dan operasi daftar API.

### Apa itu visibilitas sumber daya terkelola?

AWS layanan seperti Amazon EKS, Amazon ECS, WorkSpaces Core, AWS Lambda, dan Amazon Bedrock menyediakan dan mengoperasikan inst AgentCore ans Amazon EC2 langsung di dalam akun Anda. Layanan ini bertanggung jawab atas penskalaan, patch OS, pembaruan keamanan, dan manajemen siklus hidup. Instans Amazon EC2 yang dihasilkan, templat peluncuran Amazon EC2, volume Amazon EBS, snapshot, dan antarmuka jaringan (ENI) muncul di samping sumber daya yang dikelola pelanggan Anda di konsol dan API Amazon EC2. Setelan visibilitas sumber daya terkelola memberi Anda kontrol atas apakah sumber daya terkelola ini muncul di tampilan sumber daya Anda.

### Jenis sumber daya yang terpengaruh

| Tipe sumber daya | Layanan yang menyediakan sumber daya ini | Deskripsi |
| --- | --- | --- |
| Instans Amazon EC2 | Node pekerja Amazon EKS, instans kontainer Amazon ECS, lingkungan eksekusi AWS Lambda, Amazon | Jenis sumber daya utama dipengaruhi oleh pengaturan visibilitas |

| Tipe sumber daya | Layanan yang menyediakan sumber daya ini | Deskripsi |
| --- | --- | --- |
| | WorkSpaces Core, Amazon Bedrock AgentCore | |
| Templat peluncuran Amazon EC2 | Amazon EKS, Amazon ECS, Inti Amazon WorkSpaces , Dasar Amazon AgentCore | Luncurkan template yang dibuat oleh layanan terkelola |
| Volume Amazon EBS | Amazon EKS, Amazon ECS, Amazon WorkSpaces Core, Amazon Dasar, Peluncuran C AgentCore epat Amazon EC2 | Volume yang dilampirkan ke instans terkelola |
| Snapshot Amazon EBS | Peluncuran Cepat Amazon EC2 | Snapshot dikelola oleh Amazon EC2 Fast Launch |
| Antarmuka jaringan (ENIS) | Amazon EKS, Amazon ECS, AWS Lambda, Amazon WorkSpaces Core, Amazon Bedrock AgentCore, Pemeriksaan Status Aplikasi Amazon EC2 | Antarmuka jaringan disediakan untuk beban kerja terkelola |

> **Note**
>
> Secara default, Amazon EC2 menyembunyikan sumber daya terkelola untuk akun yang tidak memiliki sumber daya terkelola sebelum pengaturan visibilitas tersedia. Untuk akun yang sudah memiliki sumber daya terkelola, Amazon EC2 menetapkan setelan visibilitas ke Visible untuk mempertahankan alur kerja yang ada. Setelan visibilitas berlaku untuk semua sumber daya terkelola di Wilayah, terlepas dari kapan sumber daya tersebut dibuat. Anda dapat mengubah pengaturan visibilitas kapan saja.

## Mengapa mengkonfigurasi pengaturan visibilitas

Mengonfigurasi setelan visibilitas memungkinkan Anda menyesuaikan tampilan sumber daya terkelola di seluruh perkakas operasional Anda. Kasus penggunaan umum meliputi:

- Sederhanakan tata kelola dengan mengurangi jumlah sumber daya di dasbor kepatuhan menjadi hanya sumber daya yang dikelola pelanggan.
- Kurangi noise pada alat pengamatan yang menggabungkan metrik Amazon EC2 di semua instans dalam akun.
- Mencegah kesalahan positif dalam pemindai manajemen postur keamanan cloud (CSPM) (misalnya, Qualys) yang menandai sumber daya terkelola sebagai kesalahan konfigurasi pelanggan.
- Dengan instans terkelola, AWS bertanggung jawab atas konfigurasi, tambalan, dan kesehatan instans Amazon EC2. Dengan mengontrol visibilitas, Anda dapat mengartikulasikan model tanggung jawab bersama dengan lebih baik kepada pengguna akhir.

> **Note**
>
> Pengaturan visibilitas mengontrol tampilan sumber daya dalam tampilan AWS konsol dan operasi daftar API. Mereka tidak memengaruhi penagihan, operasi sumber daya, atau izin akses aktual. Sumber daya tersembunyi tetap beroperasi penuh dan dapat ditagih.

## Konfigurasikan visibilitas sumber daya terkelola

Anda dapat mengonfigurasi visibilitas sumber daya terkelola dengan menggunakan konsol Amazon EC2 atau. AWS CLI

Console

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Peng aturan, pilih Sumber daya terkelola.
4. Pilih Ubah.
5. Di bawah Visibilitas default, pilih salah satu opsi berikut:
   - Tersembunyi (default) — Menyembunyikan semua sumber daya yang dikelola.
   - Terlihat - Menampilkan semua sumber daya yang dikelola.
6. Pilih Ubah visibilitas.

AWS CLI

Dapatkan pengaturan visibilitas saat ini

Gunakan perintah get-managed-resource-visibilitas untuk mengambil konfigurasi visibilitas saat ini:

```
aws ec2 get-managed-resource-visibility
```

Contoh respons:

```
{
    "visibility": {
        "defaultVisibility": "hidden"
    }
}
```

Sembunyikan semua sumber daya yang dikelola

Gunakan perintah modify-managed-resource-visibilitas untuk menyembunyikan semua sumber daya terkelola terlepas dari operator:

```
aws ec2 modify-managed-resource-visibility \
    --default-visibility "hidden"
```

## Temukan sumber daya terkelola tersembunyi

Saat menonaktifkan visibilitas, Anda masih dapat mengakses sumber daya terkelola. Metode berikut memunculkannya sesuai permintaan:

1. Service-specific konsol: Arahkan ke konsol AWS layanan masing-masing (misalnya, konsol Amazon EKS) untuk melihat instans yang disediakan untuk layanan tersebut. Konsol layanan menyediakan detail lengkap tentang semua sumber daya yang dikelola layanan di akun Anda.
2. Kueri API langsung: Gunakan `describe-instances` API dengan `instance-id` parameter tertentu. Kueri langsung dengan ID instance yang diketahui mengembalikan hasil terlepas dari pengaturan visibilitas. Pengaturan visibilitas hanya memengaruhi operasi daftar dan filter. Anda juga dapat menggunakan `describe-instances` dengan `include-managed-resources` parameter untuk menemukan instance terkelola.

> **Note**
>
> Perilaku Direct-Query-by-ID yang sama berlaku untuk semua jenis sumber daya yang terpengaruh. Anda dapat menggunakan`describe-volumes`,`describe-launch-templates`, dan `describe-network-interfaces` dengan ID sumber daya tertentu untuk mengakses sumber daya terkelola tersembunyi dari jenis tersebut.

## Pertimbangan penagihan

Pengaturan visibilitas sumber daya terkelola tidak berpengaruh pada penagihan. Instans terkelola tersembunyi terus muncul di data penagihan karena merupakan sumber daya yang berjalan di dalam akun Anda, disediakan atas nama Anda, dan tetap dapat ditagih sepenuhnya terlepas dari konfigurasi visibilitas.

Sumber daya tersembunyi tetap terlihat di:

- AWS tagihan
- AWS Laporan Biaya dan Penggunaan

> **Important**
>
> Instans terkelola disediakan di akun Anda dan menggunakan sumber daya komputasi. Menyembunyikannya dari tampilan konsol tidak mengurangi biaya. Tinjau dokumentasi penagihan khusus layanan (misalnya, Harga Amazon EKS, Harga Amazon ECS) untuk detail tentang biaya instans terkelola.

## Batasan

- Pengaturan visibilitas berlaku untuk seluruh akun dan memengaruhi semua prinsipal IAM secara seragam.
- Anda tidak dapat secara selektif menampilkan atau menyembunyikan sumber daya terkelola berdasarkan jenis sumber daya atau layanan yang membuatnya. Misalnya, Anda tidak dapat memilih untuk menampilkan instans terkelola yang dibuat oleh Amazon EKS sambil menyembunyikan instans yang dibuat oleh Lambda, Amazon ECS, atau Amazon. WorkSpaces

## Memulai dengan instans terkelola

Untuk memulai dengan instans terkelola, lihat dokumentasi untuk penyedia layanan pilihan Anda:

- Otomatisasi infrastruktur cluster dengan Mode Otomatis EKS di Panduan Pengguna Amazon EKS
- Instans terkelola Amazon ECS di Panduan Pengembang Amazon ECS
- Menyebarkan instans terkelola di WorkSpaces Panduan Perencanaan Inti
- Instans terkelola Lambda di Panduan Pengembang Lambda
- Peluncuran Cepat Amazon EC2 dalam panduan ini
- Instans AgentCore runtime Bedrock di Panduan Pengembang Amazon Bedrock AgentCore

# Gunakan virtualisasi bersarang untuk menjalankan hypervisor di instans Amazon EC2

Virtualisasi bersarang memungkinkan Anda menjalankan hypervisor seperti Hyper-V dan KVM di dalam instans virtual Amazon EC2. Instans EC2 virtual adalah instans non-bare metal. Kemampuan ini memperluas fleksibilitas virtualisasi dengan menambahkan dukungan virtualisasi tingkat prosesor ke instans EC2 virtual, memungkinkan hypervisor yang berjalan di instans Anda untuk membuat dan mengelola mesin virtual.

Virtualisasi bersarang dapat membantu ketika Anda menjalankan alat pengembangan seperti Docker Desktop, Windows Subsistem untuk Linux 2 (WSL2), emulator Android Studio, atau QEMU dalam alur kerja pengembangan Anda, karena memungkinkan Anda memilih dari berbagai jenis instans virtual Amazon EC2 standar yang memenuhi persyaratan kinerja dan harga spesifik Anda.

Tidak ada biaya tambahan untuk menggunakan virtualisasi bersarang.

Daftar Isi


- Cara kerjanya
- Pertimbangan-pertimbangan
- Luncurkan instance baru dengan virtualisasi bersarang diaktifkan
- Mengkonfigurasi instance yang ada untuk menggunakan virtualisasi bersarang
- Temukan jenis instans yang mendukung virtualisasi bersarang

## Cara kerjanya

Instans EC2 virtual berjalan pada host fisik yang memiliki hypervisor Nitro. Untuk mendukung virtualisasi bersarang, Sistem Nitro meneruskan ekstensi prosesor, seperti Intel VT-x, ke instans untuk memfasilitasi menjalankan mesin virtual bersarang. Arsitektur virtualisasi bersarang terdiri dari tiga lapisan: AWS infrastruktur fisik dan Nitro hypervisor (L0), instans EC2 Anda menjalankan hypervisor (L1), dan satu atau lebih mesin virtual yang dibuat dalam instance tersebut (L2).

## Pertimbangan-pertimbangan

Sebelum Anda mulai menggunakan virtualisasi bersarang, pertimbangkan hal berikut:

- Jenis instans yang didukung — Virtualisasi bersarang saat ini didukung pada jenis instans berikut:
  - Tujuan Umum: M7i | | M8i M7i-flex | M8id | M8i-flex
  - Komputasi Dioptimalkan: C7i | | C8i | C C7i-flex 8id | C8i-flex
  - Memori Dioptimalkan: R7i | R7iz | R8i | R8id | | X8i R8i-flex
  - Penyimpanan Dioptimalkan: i7i | i7ie
- Hypervisor yang didukung — Saat ini, KVM dan Hyper-V merupakan hypervisor L1 yang didukung.
- Instans Windows — Saat virtualisasi bersarang diaktifkan pada instance Windows:
  - Credential Guard — Virtual Secure Mode (VSM) secara otomatis dinonaktifkan.
  - Hibernasi — Hibernasi instans dan resume tidak didukung.
  - Batas CPU — Tidak didukung pada instans Windows dengan lebih dari 192 CPU, seperti`m8i.96xl`.
- Tanggung jawab keamanan — Saat menggunakan virtualisasi bersarang pada instans EC2, AWS bertanggung jawab atas “keamanan cloud,” melindungi infrastruktur yang mendasarinya dan mempertahankan batas isolasi yang kuat antara instans EC2 yang disediakan oleh Sistem Nitro. AWS Pelanggan bertanggung jawab atas “keamanan di cloud,” yang mencakup mengamankan sistem operasi, hypervisor, mesin virtual bersarang, sistem operasi tamu, aplikasi, dan data dalam instans EC2.
- Kinerja — AWS merekomendasikan agar pelanggan yang ingin menjalankan beban kerja yang memerlukan akses ke ekstensi virtualisasi perangkat keras, dan sensitif kinerja atau memiliki persyaratan latensi yang ketat, untuk mengevaluasi instans bare metal.

# Luncurkan instance baru dengan virtualisasi bersarang diaktifkan

Saat meluncurkan instance baru, Anda dapat mengaktifkan virtualisasi bersarang untuk menjalankan hypervisor dan mesin virtual di atasnya.

Console

Untuk mengaktifkan virtualisasi bersarang selama peluncuran instance

1. Ikuti prosedur the section called “Luncurkan menggunakan wizard peluncuran instans” dan konfigurasikan instans Anda sesuai kebutuhan.
2. Pastikan jenis instans yang didukung dipilih.
3. Perluas Detail lanjutan, dan untuk virtualisasi bersarang, pilih Akti fkan.
4. Di panel Ringkasan, tinjau konfigurasi instans Anda, lalu pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan instance dengan virtualisasi bersarang diaktifkan

Gunakan perintah run-instans.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type r8i.4xlarge \
    --cpu-options "NestedVirtualization=enabled" \
    --key-name my-key-pair
```

PowerShell

Untuk meluncurkan instance dengan virtualisasi bersarang diaktifkan

Gunakan perintah New-EC2Instance.

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType r8i.4xlarge `
    -CpuOption @{NestedVirtualization='enabled'} `
    -KeyName my-key-pair
```

# Mengkonfigurasi instance yang ada untuk menggunakan virtualisasi bersarang

Anda dapat mengaktifkan virtualisasi bersarang pada instans Amazon EC2 yang sudah ada.

### Prasyarat

- Contoh harus dalam `stopped` keadaan.
- Jenis instance harus mendukung virtualisasi bersarang.

### Console

Untuk mengaktifkan virtualisasi bersarang pada instance yang ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance yang ingin Anda modifikasi dari tabel instance.
4. Pilih T ind akan, Pengaturan Inst ance, Ubah opsi CPU.
5. Pada halaman Ubah opsi CPU, untuk virtu alisasi bersar ang, pilih salah satu opsi berikut:
   - Aktif kan - Mengaktifkan virtualisasi bersarang untuk instance
   - Nonaktifkan - Mematikan virtualisasi bersarang untuk instance
6. Tinjau perubahan Anda, lalu pilih Ubah untuk menerapkan opsi CPU baru.

### AWS CLI

Untuk mengaktifkan virtualisasi bersarang pada instance yang ada

Pertama hentikan instance, dan kemudian gunakan perintah modify- instance-cpu-options.

```
aws ec2 modify-instance-cpu-options \
    --instance-id i-1234567890abcdef0 \
    --core-count 4 \
    --threads-per-core 2 \
    --nested-virtualization enabled
```

PowerShell

Untuk mengaktifkan virtualisasi bersarang pada instance yang ada

Pertama hentikan instance, dan kemudian gunakan Edit-EC2InstanceCpuOption perintah.

```
Edit-EC2InstanceCpuOption `
    -InstanceId i-1234567890abcdef0 `
    -CoreCount 4 `
    -ThreadsPerCore 2 `
    -NestedVirtualization enabled
```

## Temukan jenis instans yang mendukung virtualisasi bersarang

Anda dapat menemukan jenis instans Amazon EC2 mana yang mendukung virtualisasi bersarang. Anda dapat melakukan query di seluruh Wilayah AWS atau memeriksa jenis instance tertentu.

AWS CLI

Untuk membuat daftar semua keluarga instans yang mendukung virtualisasi bersarang di Wilayah

Gunakan perintah describe-instance-types dengan `processor-info.supported-features` filter disetel ke nilai: `nested-virtualization`

```
aws ec2 describe-instance-types --region us-east-1 \
  --filters "Name=processor-info.supported-features,Values=nested-virtualization" \
  --query "InstanceTypes[].InstanceType" --output text \
  | tr '\t' '\n' | cut -d. -f1 | sort -u
```

Untuk memeriksa apakah jenis instance tertentu mendukung virtualisasi bersarang

Gunakan perintah describe-instance-types dan kueri `ProcessorInfo.SupportedFeatures` bidang jenis instance yang ditentukan:

```
aws ec2 describe-instance-types \
  --instance-types m8i.2xlarge \
  --query "InstanceTypes[].ProcessorInfo.SupportedFeatures"
```

Jika jenis instance mendukung virtualisasi bersarang, output termasuk`nested-virtualization`. Jika tidak, perintah mengembalikan array kosong (`[]`).

PowerShell

Untuk membuat daftar semua keluarga instans yang mendukung virtualisasi bersarang di Wilayah

Gunakan Get-EC2InstanceType perintah dengan `processor-info.supported-features` filter diatur ke nilai`nested-virtualization`:

```
Get-EC2InstanceType -Region us-east-1 -Filter @{ Name='processor-info.supported-
features'; Values='nested-virtualization' } |
  ForEach-Object { ($_.InstanceType -split '\.')[0] } |
  Sort-Object -Unique
```

Untuk memeriksa apakah jenis instance tertentu mendukung virtualisasi bersarang

Gunakan Get-EC2InstanceType perintah dan kueri `ProcessorInfo.SupportedFeatures` properti dari jenis instance yang ditentukan:

```
(Get-EC2InstanceType -InstanceType m8i.2xlarge).ProcessorInfo.SupportedFeatures
```

Jika jenis instance mendukung virtualisasi bersarang, output termasuk`nested-virtualization`. Jika tidak, perintah tidak mengembalikan output.

# Opsi penagihan dan pembelian Amazon EC2

Anda dapat menggunakan opsi berikut untuk mengoptimalkan biaya Anda untuk Amazon EC2:

- Instans Sesuai Permintaan – Bayar, per detik, untuk instans yang Anda luncurkan.
- Savings Plans – Kurangi biaya Amazon EC2 dengan membuat komitmen pada jumlah penggunaan yang konsisten, dalam USD per jam, untuk jangka waktu 1 atau 3 tahun.
- Instans Terpesan – Kurangi biaya Amazon EC2 Anda dengan membuat komitmen pada konfigurasi instans yang konsisten, termasuk tipe instans dan Wilayah, untuk jangka waktu 1 atau 3 tahun.
- Instans Spot – Minta instans EC2 yang tidak digunakan, yang dapat mengurangi biaya Amazon EC2 Anda secara signifikan.
- Host Khusus - Bayar untuk host fisik yang sepenuhnya didedikasikan untuk menjalankan instans Anda, dan bawa lisensi perangkat lunak per soket, per inti, atau per VM yang ada untuk mengurangi biaya.

- Instans Khusus - Bayar, per jam, untuk instans yang berjalan pada perangkat keras penghuni tunggal.
- Reservasi Kapasitas — Kapasitas cadangan untuk instans EC2 Anda di Availability Zone tertentu.

Jika Anda tidak dapat membuat komitmen terhadap konfigurasi instans tertentu, tetapi Anda dapat berkomitmen pada jumlah penggunaan, belilah Savings Plans untuk mengurangi biaya Instans Sesuai Permintaan.Jika Anda memerlukan reservasi kapasitas, belilah Instans Terpesan atau Reservasi Kapasitas untuk Zona Ketersediaan tertentu. Blok Kapasitas dapat digunakan untuk memesan klaster instans GPU. Instans Spot adalah pilihan hemat biaya jika Anda dapat bersikap fleksibel tentang kapan aplikasi Anda berjalan dan apakah aplikasi tersebut dapat diinterupsi. Host Khusus atau Instans Khusus dapat membantu Anda memenuhi persyaratan kepatuhan dan mengurangi biaya dengan menggunakan lisensi perangkat lunak terikat server yang ada.

Untuk informasi selengkapnya, lihat Harga Amazon EC2 dan. Instans terkelola Amazon EC2

## Membeli Instans Sesuai Permintaan untuk Amazon EC2

Dengan Instans Sesuai Permintaan, Anda membayar kapasitas komputasi per detik tanpa komitmen jangka panjang. Anda memiliki kendali penuh atas siklus hidup instans—Anda memutuskan kapan akan meluncurkan, menghentikan, hibernasi, memulai, melakukan reboot, atau mengakhirinya.

Tidak ada komitmen jangka panjang yang diperlukan saat Anda membeli Instans Sesuai Permintaan. Anda hanya membayar untuk detik saat Instans Sesuai Permintaan Anda berada pada status `running`, dengan minimum 60 detik. Harga per detik untuk Instans Sesuai Permintaan yang berjalan ditetapkan, dan tercantum di halaman EC2 Harga Amazon, Harga Sesuai Permintaan, halaman .

Kami menyarankan agar Anda menggunakan Instans Sesuai Permintaan untuk aplikasi dengan beban kerja tidak teratur jangka pendek yang tidak dapat diganggu.

Untuk penghematan Instans Sesuai Permintaan yang signifikan, gunakan AWS Savings Plans, Instans Spot, atau Ikhtisar Instans Cadangan untuk Amazon EC2.

Daftar Isi


- Kuota Instans Sesuai Permintaan
  - Memantau kuota dan penggunaan Instans Sesuai Permintaan
  - Meminta peningkatan kuota
- Membuat kueri harga Instans Sesuai Permintaan

## Kuota Instans Sesuai Permintaan

Ada kuota untuk jumlah Instans Sesuai Permintaan yang berjalan per Akun AWS Wilayah. Kuota Instans Sesuai Permintaan dikelola berdasarkan jumlah unit pemrosesan pusat virtual (vCPUs) yang digunakan Instans Sesuai Permintaan Anda, apa pun jenis instancenya. Setiap jenis kuota menentukan jumlah maksimum v CPUs untuk satu atau lebih keluarga instance.

Akun Anda menyertakan kuota berikut untuk Instans Sesuai Permintaan. Instans yang berada dalam status tertunda, berhenti, berhenti, dan hibernasi tidak dihitung dalam kuota Instans Sesuai Permintaan Anda. Reservasi Kapasitas dihitung dalam kuota Instans Sesuai Permintaan Anda, meskipun tidak digunakan.

| Nama | Default | Dapat disesuaikan |
|---|---|---|
| Instans DL Sesuai Permintaan yang Berjalan | 0 | Ya |
| Instans F Sesuai Permintaan yang Berjalan | 0 | Ya |
| Instans DL Sesuai Permintaan yang Berjalan dan instans VT | 0 | Ya |
| Instans HPC Sesuai Permintaan yang Berjalan | 0 | Ya |
| Instans Memori Tinggi Sesuai Permintaan yang Berjalan | 0 | Ya |
| Instans Inf Sesuai Permintaan yang Berjalan | 0 | Ya |
| Instans P Sesuai Permintaan yang Berjalan | 0 | Ya |
| Instans Standar (A, C, D, H, I, M, R, T, Z) Sesuai Permintaan yang Berjalan | 5 | Ya |
| Instans Trn Sesuai Permintaan yang Berjalan | 0 | Ya |
| Instans X Sesuai Permintaan yang Berjalan | 0 | Ya |

Untuk informasi tentang keluarga, generasi, dan ukuran instans yang berbeda, lihat Panduan Jenis EC2 Instans Amazon.

Anda dapat meluncurkan kombinasi jenis instans yang memenuhi kebutuhan aplikasi yang berubah, selama jumlah v CPUs tidak melebihi kuota akun Anda. Misalnya, dengan kuota instans Standar 256 vCPUs, Anda dapat meluncurkan 32 `m5.2xlarge` instance (32 x 8 vCPUs) atau 16 `c5.4xlarge` instance (16 x 16 v). CPUs Untuk informasi selengkapnya, lihat Batas Instans EC2 Sesuai Permintaan.

Tugas

- Memantau kuota dan penggunaan Instans Sesuai Permintaan
- Meminta peningkatan kuota

Memantau kuota dan penggunaan Instans Sesuai Permintaan

Anda dapat melihat dan mengelola kuota Instans Sesuai Permintaan untuk setiap Wilayah menggunakan metode berikut.

Untuk melihat kuota saat ini menggunakan konsol Kuota Layanan

1. Buka konsol Service Quotas di https://console.aws.amazon.com/servicequotas/home/services/ec2/quotas/.
2. Dari bilah navigasi, pilih Wilayah.
3. Di bidang filter, masukkan **`On-Demand`**.
4. Kolom Nilai kuota Terapan menampilkan jumlah maksimum v CPUs untuk setiap jenis kuota Instans Sesuai Permintaan untuk akun Anda.

Untuk melihat kuota Anda saat ini menggunakan konsol AWS Trusted Advisor

Buka halaman batas layanan di AWS Trusted Advisor konsol.

Untuk mengkonfigurasi CloudWatch alarm

Dengan integrasi CloudWatch metrik Amazon, Anda dapat memantau EC2 penggunaan Anda terhadap kuota Anda. Anda juga dapat mengonfigurasi alarm untuk memperingatkan saat sudah mendekati kuota. Untuk informasi selengkapnya, lihat Service Quotas dan CloudWatch alarm Amazon di Panduan Pengguna Service Quotas.

Meminta peningkatan kuota

Meskipun Amazon EC2 secara otomatis meningkatkan kuota Instans Sesuai Permintaan berdasarkan penggunaan, Anda dapat meminta peningkatan kuota jika perlu. Misalnya, jika Anda

bermaksud untuk meluncurkan lebih banyak instans daripada yang diizinkan oleh kuota saat ini, Anda dapat meminta peningkatan kuota dengan menggunakan Konsol Kuota Layanan yang dijelaskan di Kuota layanan Amazon EC2.

### Membuat kueri harga Instans Sesuai Permintaan

Anda dapat menggunakan API Layanan Daftar Harga atau API Daftar AWS Harga untuk menanyakan harga Instans Sesuai Permintaan. Untuk informasi selengkapnya, lihat Menggunakan API Daftar AWS Harga di Panduan AWS Billing Pengguna.

## Ikhtisar Instans Cadangan untuk Amazon EC2

> **Important**
>
> Kami merekomendasikan Paket Tabungan daripada Instans Cadangan. Paket Tabungan adalah cara termudah dan paling fleksibel untuk menghemat biaya AWS komputasi Anda dan menawarkan harga yang lebih rendah (diskon hingga 72% dari On-Demand harga), sama seperti Instans Cadangan. Namun, Paket Tabungan berbeda dengan Instans Cadangan. Dengan Instans Cadangan, Anda membuat komitmen untuk konfigurasi instans tertentu, sedangkan dengan Paket Tabungan, Anda memiliki fleksibilitas untuk menggunakan konfigurasi instans yang paling sesuai dengan kebutuhan Anda. Untuk menggunakan Paket Tabungan, Anda membuat komitmen untuk jumlah penggunaan yang konsisten, diukur dalam USD per jam. Untuk informasi selengkapnya, lihat AWS Panduan Pengguna Savings Plans.

Instans Cadangan memberi Anda penghematan yang signifikan untuk biaya Amazon EC2 dibandingkan dengan harga On-Demand Instans. Instans Cadangan bukan instans fisik, melainkan diskon penagihan yang diterapkan pada penggunaan On-Demand Instans di akun Anda. On-Demand Instans ini harus sesuai dengan atribut tertentu, seperti jenis instans dan Wilayah, untuk mendapatkan manfaat dari diskon penagihan.

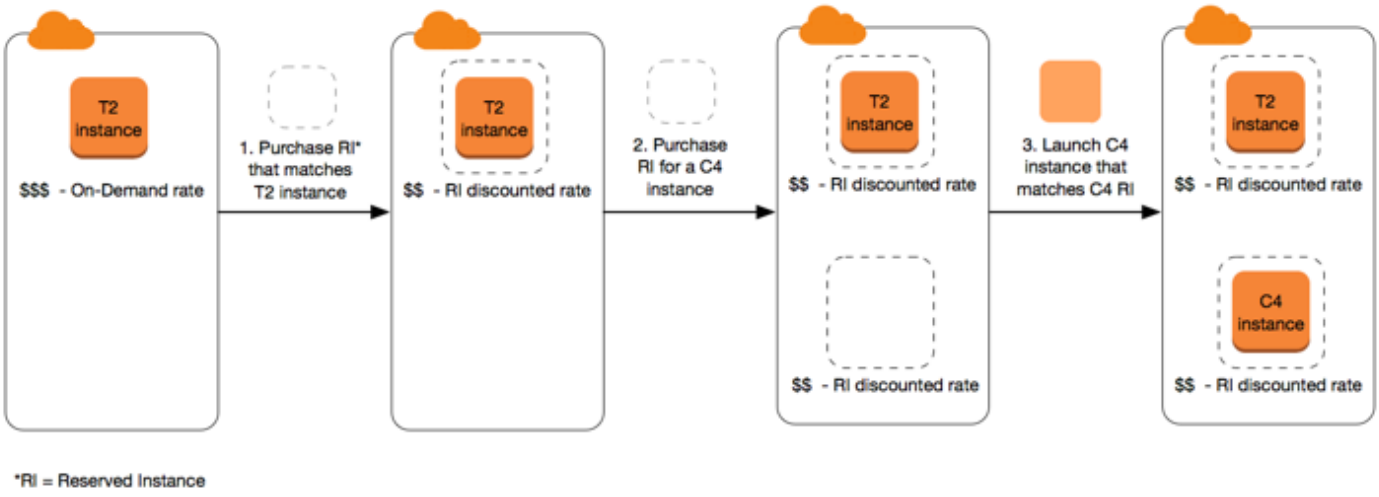
Topik Instans Terpesan

- Contoh skenario Instance Cadangan
- Variabel utama yang menentukan harga Instans Terpesan
- Instans Terpesan Regional dan zonal (cakupan)
- Tipe Instans Terpesan (kelas penawaran)
- Bagaimana diskon Instans Cadangan diterapkan

- Menggunakan Instans Terpesan Anda
- Cara kerja penagihan dengan Instans Cadangan
- Beli Instans Cadangan untuk Amazon EC2
- Jual Instans Cadangan untuk Amazon EC2 di Marketplace Instans Cadangan
- Memodifikasi Instans Terpesan
- Menukar Instans Terpesan Konvertibel
- Kuota Instans Terpesan

## Contoh skenario Instance Cadangan

Diagram berikut menunjukkan skenario dasar pembelian dan penggunaan Instans Cadangan.




Dalam skenario ini, Anda memiliki Inst On-Demand ans (T2) yang sedang berjalan di akun Anda, yang saat ini Anda bayar On-Demand tarifnya. Anda membeli Instans Terpesan yang cocok dengan atribut instans Anda yang sedang berjalan, dan manfaat penagihan segera diterapkan. Selanjutnya, Anda membeli Instans Terpesan untuk instans C4. Anda tidak memiliki instans yang sedang berjalan di akun Anda yang cocok dengan atribut Instans Terpesan ini. Pada langkah terakhir, Anda meluncurkan instans yang cocok dengan atribut Instans Terpesan C4, dan manfaat penagihan segera diterapkan.

## Variabel utama yang menentukan harga Instans Terpesan

Harga Instans Terpesan ditentukan oleh variabel kunci berikut.

### Atribut instans

Instans Terpesan memiliki empat atribut instans yang menentukan harganya.

- Tipe instans: Contohnya, `m4.large`. Ini terdiri dari keluarga instans (sebagai contoh, `m4`) dan ukuran instans (misalnya, `large`).
- Wilayah: Wilayah tempat Instans Terpesan dibeli.
- Penghunian: Apakah instans Anda berjalan pada perangkat keras bersama (default) atau penghuni tunggal (khusus). Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2.
- Platform: Sistem operasi; misalnya, Windows atau Linux/Unix. Untuk informasi selengkapnya, lihat Memilih platform.

### Komitmen jangka waktu

Anda dapat membeli Instans Terpesan untuk komitmen satu tahun atau tiga tahun, di mana komitmen tiga tahun menawarkan diskon yang lebih besar.

- One-year: Satu tahun didefinisikan sebagai 31536000 detik (365 hari).
- Three-yearTiga tahun didefinisikan sebagai 94608000 detik (1095 hari).

Instans Cadangan tidak diperpanjang secara otomatis; saat kedaluwarsa, Anda dapat terus menggunakan instans EC2 tanpa gangguan, tetapi Anda akan dikenakan On-Demand tarif. Dalam contoh di atas, ketika Instans Cadangan yang mencakup instans T2 dan C4 kedaluwarsa, Anda kembali membayar On-Demand tarif sampai Anda menghentikan instans atau membeli Instans Cadangan baru yang cocok dengan atribut instance.

> **Important**
>
> Setelah Anda membeli Instans Terpesan, Anda tidak dapat membatalkan pembelian tersebut. Namun, Anda mungkin dapat mengubah, menukar, atau menjual Instans Terpesan itu jika kebutuhan Anda berubah.

### Opsi pembayaran

Opsi pembayaran berikut tersedia untuk Instans Terpesan:

- Lunas di Depan: Pembayaran penuh dilakukan di awal jangka waktu, tanpa biaya lain atau biaya per jam tambahan yang timbul untuk sisa jangka waktu, berapa pun jam yang digunakan.
- Dengan Uang Muka: Sebagian dari biaya harus dibayar di muka dan sisa jam dalam jangka waktu tersebut ditagih dengan tarif per jam yang didiskon, terlepas dari apakah Instans Terpesan tersebut sedang digunakan.
- Tanpa Uang Muka: Anda akan dikenai tarif per jam dengan diskon untuk setiap jam dalam jangka waktu tersebut, terlepas dari apakah Instans Terpesan sedang digunakan. Tidak perlu uang muka.

> **Note**
>
> Tidak ada Instans Terpesan Tanpa Uang Muka yang didasarkan pada kewajiban kontraktual untuk membayar bulanan untuk seluruh jangka waktu reservasi. Untuk alasan ini, riwayat penagihan yang berhasil diperlukan sebelum Anda dapat membeli Instans Terpesan Tanpa Uang Muka.

Secara umum, Anda dapat menghemat lebih banyak uang dengan membayar uang muka yang lebih tinggi untuk Instans Terpesan. Anda juga dapat menemukan Instans Terpesan yang ditawarkan oleh penjual pihak ketiga dengan harga lebih rendah dan jangka waktu lebih pendek di Pasar Instans Terpesan. Untuk informasi selengkapnya, lihat Jual Instans Cadangan untuk Amazon EC2 di Marketplace Instans Cadangan.

Kelas penawaran

Jika kebutuhan komputasi Anda berubah, Anda mungkin dapat mengubah atau menukar Instans Cadangan Anda, bergantung pada kelas penawaran.

- Standar: Kelas ini memberikan diskon paling signifikan, tetapi hanya dapat dimodifikasi. Instans Terpesan Standar tidak dapat ditukar.
- Konvertibel: Kelas ini memberikan diskon yang lebih rendah daripada Instans Terpesan Standar, tetapi dapat ditukar dengan Instans Terpesan Konvertibel lainnya dengan atribut instans yang berbeda. Instans Terpesan Konvertibel juga dapat dimodifikasi.

Untuk informasi selengkapnya, lihat Tipe Instans Terpesan (kelas penawaran).

> **Important**
>
> Setelah Anda membeli Instans Terpesan, Anda tidak dapat membatalkan pembelian tersebut. Namun, Anda mungkin dapat mengubah, menukar, atau menjual Instans Terpesan itu jika kebutuhan Anda berubah.

Untuk informasi selengkapnya, lihat halaman Harga Instans Terpesan Amazon EC2.

## Instans Terpesan Regional dan zonal (cakupan)

Saat Anda membeli Instans Terpesan, Anda menentukan cakupan Instans Terpesan tersebut. Cakupan itu bisa regional atau zonal.

- Regional: Saat Anda membeli Instans Terpesan untuk suatu Wilayah, maka instans itu disebut sebagai Instans Terpesan regional.
- Zonal: Saat Anda membeli Instans Terpesan untuk Zona Ketersediaan tertentu, instans itu disebut sebagai Instans Terpesan zonal.

Cakupan tidak memengaruhi harga. Anda membayar harga yang sama untuk Instans Terpesan regional maupun zonal. Untuk informasi selengkapnya tentang harga Instans Terpesan, lihat Variabel utama yang menentukan harga Instans Terpesan dan Harga Instans Terpesan Amazon EC2.

Untuk informasi selengkapnya tentang cara menentukan cakupan Instans Terpesan, lihat Atribut RI, khususnya bullet Zona Ketersediaan.

Perbedaan antara Instans Terpesan regional dan zonal

Tabel berikut menyoroti beberapa perbedaan utama antara Instans Terpesan regional dan Instans Terpesan zonal:

| | Instans Terpesan Regional | Instans Terpesan Zonal |
|---|---|---|
| Kemampuan untuk memesan kapasitas | Instans Terpesan regional tidak memesan kapasitas. | Instans Terpesan zonal memesan kapasitas di Zona Ketersediaan yang ditentukan. |

| | Instans Terpesan Regional | Instans Terpesan Zonal |
|---|---|---|
| Fleksibilitas Zona Ketersediaan | Diskon Instans Terpesan berlaku untuk penggunaan instans di Zona Ketersediaan mana pun di Wilayah yang ditentukan. | Tidak ada fleksibilitas Zona Ketersediaan — diskon Instans Terpesan berlaku untuk penggunaan instans di Zona Ketersediaan yang ditentukan saja. |
| Fleksibilitas ukuran instans | Diskon Instans Terpesan berlaku untuk penggunaan instans dalam keluarga instans, berapa pun ukurannya.<br><br>Hanya didukung di Instans Linux/Unix Cadangan Amazon dengan sewa default. Untuk informasi selengkapnya, lihat Fleksibilitas ukuran instans ditentukan oleh faktor normalisasi. | Tidak ada fleksibilitas ukuran instans—Diskon Instans Cadangan hanya berlaku untuk penggunaan instans untuk keluarga instans dan ukuran instans yang ditentukan. |
| Mengantrekan pembelian | Anda dapat mengantrekan pembelian untuk Instans Terpesan regional. | Anda dapat mengantrekan pembelian untuk Instans Terpesan zonal. |

Untuk informasi dan contoh selengkapnya, lihat Bagaimana diskon Instans Cadangan diterapkan.

## Tipe Instans Terpesan (kelas penawaran)

Kelas penawaran Instans Terpesan adalah Standar atau Konvertibel. Instans Terpesan Standar memberikan diskon yang lebih signifikan daripada Instans Terpesan Konvertibel, tetapi Anda tidak dapat menukarkan Instans Terpesan Standar. Anda dapat menukar Instans Terpesan Konvertibel. Anda dapat memodifikasi Instans Terpesan Standar dan Konvertibel.

Konfigurasi Instans Terpesan terdiri dari satu tipe instans, platform, cakupan, dan penghunian selama jangka waktu tertentu. Jika kebutuhan komputasi Anda berubah, Anda mungkin dapat mengubah atau menukar Instans Terpesan Anda.

Perbedaaan antara Instans Terpesan Standar dan Konvertibel

Berikut adalah perbedaaan antara Instans Terpesan Standar dan Konvertibel.

| | Instans Terpesan Standar | Instans Terpesan Konvertibel |
|---|---|---|
| Memodifikasi Instans Terpesan | Beberapa atribut dapat dimodifikasi. Untuk informasi selengkapnya, lihat Memodifikasi Instans Terpesan. | Beberapa atribut dapat dimodifikasi. Untuk informasi selengkapnya, lihat Memodifikasi Instans Terpesan. |
| Menukar Instans Terpesan | Tidak bisa ditukar. | Dapat ditukar selama jangka waktu dengan Instans Terpesan Konvertibel lainnya dengan atribut baru, termasuk keluarga instans, tipe instans, platform, cakupan, atau penghunian. Untuk informasi selengkapnya, lihat Menukar Instans Terpesan Konvertibel. |
| Menjual di Marketplace Instans Terpesan | Dapat dijual di Marketplace Instans Terpesan. | Tidak dapat dijual di Marketplace Instans Terpesan. |
| Membeli dari Marketplace Instans Terpesan | Dapat dibeli di Marketplace Instans Terpesan. | Tidak dapat dibeli di Marketplace Instans Terpesan. |

## Bagaimana diskon Instans Cadangan diterapkan

Instans Cadangan bukan instans fisik, melainkan diskon penagihan yang diterapkan pada On-Demand Instans yang sedang berjalan di akun Anda. On-Demand Instans harus sesuai dengan spesifikasi tertentu dari Instans Cadangan untuk mendapatkan keuntungan dari diskon penagihan.

Jika Anda membeli Instans Cadangan dan Anda sudah memiliki Inst On-Demand ans yang berjalan yang sesuai dengan spesifikasi Instans Cadangan, diskon penagihan diterapkan segera dan

otomatis. Anda tidak perlu memulai ulang instans Anda. Jika Anda tidak memiliki Inst On-Demand ans berjalan yang memenuhi syarat, luncurkan On-Demand Instans dengan spesifikasi yang sama dengan Instans Cadangan Anda. Untuk informasi selengkapnya, lihat Menggunakan Instans Terpesan Anda.

Kelas penawaran (Standar atau Konvertibel) dari Instans Terpesan tidak memengaruhi bagaimana diskon penagihan diterapkan.

Topik

- Bagaimana Instans Terpesan zonal diterapkan
- Bagaimana Instans Terpesan regional diterapkan
- Fleksibilitas ukuran instans
- Contoh penerapan Instans Terpesan

### Bagaimana Instans Terpesan zonal diterapkan

Instans Terpesan yang dibeli untuk memesan kapasitas di Zona Ketersediaan tertentu disebut Instans Terpesan zonal.

- Diskon Instans Terpesan berlaku untuk penggunaan instans yang sesuai di Zona Ketersediaan tersebut.
- Atribut (sewa, platform, Zona Ketersediaan, keluarga instans, dan ukuran instans) dari instans yang sedang berjalan harus sesuai dengan Instans Cadangan.

Misalnya, jika Anda membeli dua Instans Cadangan Linux/Unix Standar sewa `c4.xlarge` default untuk Availability Zone us-east-1a, maka hingga dua instans sewa `c4.xlarge` default yang berjalan di Availability Zone us-east-1a dapat memperoleh manfaat dari diskon Linux/Unix Instans Cadangan.

### Bagaimana Instans Terpesan regional diterapkan

Instans Terpesan yang dibeli untuk suatu Wilayah disebut Instans Terpesan regional, dan menyediakan Zona Ketersediaan serta fleksibilitas ukuran instans.

- Diskon Instans Terpesan berlaku untuk penggunaan instans di Zona Ketersediaan mana pun di Wilayah tersebut.
- Diskon Instans Terpesan berlaku untuk penggunaan instans dalam keluarga instans, berapa pun ukurannya—ini dikenal sebagai fleksibilitas ukuran instans.

### Fleksibilitas ukuran instans

Dengan fleksibilitas ukuran instans, diskon Instans Cadangan berlaku untuk penggunaan instans untuk instans yang memiliki keluarga yang sama. Instans Terpesan diterapkan dari ukuran instans terkecil hingga terbesar dalam keluarga instans berdasarkan faktor normalisasi. Untuk contoh bagaimana diskon Instans Terpesan diterapkan, lihat Skenario 2: Instans Terpesan dalam satu akun menggunakan faktor normalisasi.

### Batasan

- Didukung: Fleksibilitas ukuran instans hanya didukung untuk Instans Terpesan Regional.
- Tidak didukung: Fleksibilitas ukuran instans tidak didukung untuk Instans Terpesan berikut:
  - Instans Terpesan yang dibeli untuk Zona Ketersediaan tertentu (Instans Terpesan zonal)
  - Instans Cadangan untuk instans G4ad, G4dn, G5, G5g, G6, G6e, G6f, Gr6, Gr6f, hpc7a, P5, Inf1, Inf2, u7i-6tb, dan u7i-8tb
  - Instans Terpesan untuk Windows Server, Windows Server dengan SQL Standard, Windows Server dengan SQL Server Enterprise, Windows Server dengan SQL Server Web, RHEL, dan SUSE Linux Enterprise Server
  - Instans Terpesan dengan penghunian khusus

### Fleksibilitas ukuran instans ditentukan oleh faktor normalisasi

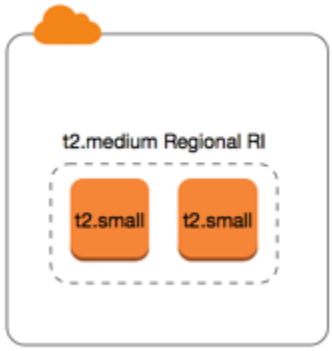
Fleksibilitas ukuran instans ditentukan oleh faktor normalisasi ukuran instans. Diskon berlaku baik sepenuhnya atau sebagian untuk instans yang berjalan dari keluarga instans yang sama, bergantung pada ukuran instans reservasi, di Zona Ketersediaan mana pun di Wilayah itu. Atribut yang harus dicocokkan adalah keluarga instans penghunian, dan platform.

Tabel berikut mencantumkan berbagai ukuran dalam keluarga instans, dan faktor normalisasi yang sesuai. Skala ini digunakan untuk menerapkan tarif diskon dari Instans Terpesan ke penggunaan normal keluarga instans.

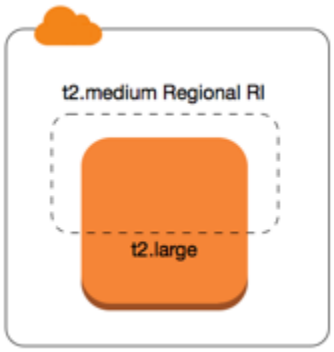
| Ukuran instans | Faktor normalisasi |
|---|---|
| nano | 0.25 |
| micro | 0.5 |
| small | 1 |

| Ukuran instans | Faktor normalisasi |
| --- | --- |
| medium | 2 |
| large | 4 |
| xlarge | 8 |
| 2xlarge | 16 |
| 3xlarge | 24 |
| 4xlarge | 32 |
| 6xlarge | 48 |
| 8xlarge | 64 |
| 9xlarge | 72 |
| 10xlarge | 80 |
| 12xlarge | 96 |
| 16xlarge | 128 |
| 18xlarge | 144 |
| 24xlarge | 192 |
| 32xlarge | 256 |
| 48xlarge | 384 |
| 56xlarge | 448 |
| 96xlarge | 768 |
| 112xlarge | 896 |

Misalnya, instans `t2.medium` memiliki faktor normalisasi 2. Jika Anda membeli Instans `t2.medium` Cadangan Amazon Linux/Unix sewa default di AS Timur (Virginia Utara) dan Anda memiliki dua `t2.small` instans yang sedang berjalan di akun Anda di Wilayah tersebut, manfaat penagihan diterapkan sepenuhnya ke kedua instans.




Atau, jika Anda memiliki satu instans `t2.large` yang berjalan di akun Anda di Wilayah AS Timur (Virginia Utara), manfaat penagihan diterapkan ke 50% penggunaan instans.




Faktor normalisasi juga diterapkan saat memodifikasi Instans Terpesan. Untuk informasi selengkapnya, lihat Memodifikasi Instans Terpesan.

Faktor normalisasi untuk instans bare metal

Fleksibilitas ukuran instans juga berlaku untuk instans bare metal dalam keluarga instans. Jika Anda memiliki Instans Cad Linux/Unix angan Amazon regional dengan sewa bersama pada instans bare metal, Anda dapat memanfaatkan penghematan Instans Cadangan dalam rangkaian instans yang sama. Kebalikannya juga benar: jika Anda memiliki Instans Cad Linux/Unix angan Amazon regional dengan sewa bersama pada instans dalam keluarga yang sama dengan instans bare metal, Anda dapat memanfaatkan penghematan Instans Cadangan pada instans bare metal.

Ukuran instans `metal` tidak memiliki faktor normalisasi tunggal. Instans bare metal memiliki faktor normalisasi yang sama dengan ukuran instans virtual setara dalam keluarga instans yang sama. Misalnya, instans `i3.metal` memiliki faktor normalisasi yang sama dengan instans `i3.16xlarge`.

| Ukuran instans | Faktor normalisasi |
|---|---|
| a1.metal | 32 |
| m5zn.metal \| x2iezn.metal z1d.metal | 96 |
| c6g.metal \| c6gd.metal \| i3.metal \| m6g.metal \| m6gd.metal \| r6g.metal \| r6gd.metal \| x2gd.metal | 128 |
| c5n.metal | 144 |
| c5.metal \| c5d.metal \| i3en.metal \| m5.metal \| m5d.metal \| m5dn.metal \| m5n.metal \| r5.metal \| r5b.metal \| r5d.metal \| r5dn.metal \| r5n.metal | 192 |
| c6i.metal \| c6id.metal \| m6i.metal \| m6id.metal \| r6d.metal \| r6id.metal | 256 |
| u-18tb1.metal \| u-24tb1.metal | 448 |
| u-6tb1.metal \| u-9tb1.metal \| u-12tb1.metal | 896 |

Misalnya, instans `i3.metal` memiliki faktor normalisasi 128. Jika Anda membeli Instans Cadangan Amazon Linux/Unix sewa `i3.metal` default di AS Timur (Virginia Utara), manfaat penagihan dapat berlaku sebagai berikut:

- Jika Anda memiliki satu `i3.16xlarge` yang berjalan di akun Anda di Wilayah tersebut, keuntungan penagihan diterapkan secara penuh ke instans `i3.16xlarge` (faktor normalisasi `i3.16xlarge` = 128).
- Atau, jika Anda memiliki dua instans `i3.8xlarge` yang berjalan di akun Anda di Wilayah tersebut, keuntungan penagihan diterapkan secara penuh ke kedua instans `i3.8xlarge` (faktor normalisasi `i3.8xlarge` = 64).

- Atau, jika Anda memiliki empat instans `i3.4xlarge` yang berjalan di akun Anda di Wilayah tersebut, keuntungan penagihan diterapkan secara penuh ke semua empat instans `i3.4xlarge` (faktor normalisasi `i3.4xlarge` = 32).

Kebalikannya juga benar. Misalnya, jika Anda membeli dua Instans Cadangan Amazon Linux/Unix sewa `i3.8xlarge` default di AS Timur (Virginia Utara), dan Anda memiliki satu `i3.metal` instans yang berjalan di Wilayah tersebut, manfaat penagihan diterapkan secara penuh ke `i3.metal` instans tersebut.

Contoh penerapan Instans Terpesan

Skenario berikut membahas cara penerapan Instans Terpesan.

- Skenario 1: Instans Terpesan dalam satu akun
- Skenario 2: Instans Terpesan dalam satu akun menggunakan faktor normalisasi
- Skenario 3: Instans Terpesan Regional dalam akun tertaut
- Skenario 4: Instans Terpesan Zonal dalam akun tertaut

Skenario 1: Instans Terpesan dalam satu akun

Anda menjalankan On-Demand Instans berikut di akun A:

- 4 x `m3.large` Linux, instans penghunian default di Zona Ketersediaan us-east-1a
- 2 x `m4.xlarge` Amazon Linux, instans penghunian default di Zona Ketersediaan us-east-1b
- 1 x `c4.xlarge` Amazon Linux, instans penghunian default di Zona Ketersediaan us-east-1c

Anda membeli Instans Terpesan berikut di akun A:

- 4 x `m3.large` Linux, Instans Terpesan penghunian default di Zona Ketersediaan us-east-1a (kapasitas dipesan)
- 4 x `m4.large` Amazon Linux, Instans Terpesan penghunian default di Wilayah us-east-1
- 1 x `c4.large` Amazon Linux, Instans Terpesan penghunian default di Wilayah us-east-1

Keuntungan Instans Terpesan diterapkan dengan cara berikut:

- Diskon dan reservasi kapasitas keempat Instans Terpesan zonal `m3.large` digunakan oleh empat instans `m3.large` karena atribut (ukuran instans, Wilayah, platform,penghunian) di antara keempatnya cocok.
- Instans Terpesan regional `m4.large` memberikan Zona Ketersediaan dan fleksibilitas ukuran instans, karena ini adalah Instans Terpesan Amazon Linux regional dengan penghunian default.

  A `m4.large` setara dengan 4 yang dinormalisasiunits/hour.

  Anda telah membeli empat Inst `m4.large` ans Cadangan regional, dan secara total, mereka sama dengan 16 Instans yang dinormalisasi units/hour (4x4). Akun A memiliki dua `m4.xlarge` instance yang berjalan, yang setara dengan 16 yang dinormalisasi units/hour (2x8). Dalam hal ini, empat Instans Terpesan regional `m4.large` memberikan manfaat penagihan penuh untuk penggunaan kedua instans `m4.xlarge`.
- Instans Terpesan regional `c4.large` di us-east-1 memberikan Zona Ketersediaan dan fleksibilitas ukuran instans, karena ini adalah Instans Terpesan Amazon Linux regional dengan penghunian default, dan diterapkan untuk instans `c4.xlarge`. Sebuah `c4.large` contoh setara dengan 4 yang dinormalisasi units/hour dan a `c4.xlarge` setara dengan 8 yang dinormalisasi units/hour.

  Dalam hal ini, Instans Terpesan regional `c4.large` memberikan sebagian keuntungan untuk penggunaan `c4.xlarge`. Ini karena Instans Cad `c4.large` angan setara dengan 4 penggunaan yang dinormal units/hour isasi, tetapi `c4.xlarge` instans memerlukan 8 yang dinormalisasiunits/hour. Oleh karena itu, diskon penagihan Instans Terpesan `c4.large` berlaku untuk 50% dari penggunaan `c4.xlarge`. Sisa pem `c4.xlarge` akaian dibebankan sesuai On-Demand tarif.

Skenario 2: Instans Terpesan dalam satu akun menggunakan faktor normalisasi

Anda menjalankan On-Demand Instans berikut di akun A:

- 2 x `m3.xlarge` Amazon Linux, instans penghunian default di Zona Ketersediaan us-east-1b
- 2 x `m3.large` Amazon Linux, instans penghunian default di Zona Ketersediaan us-east-1b

Anda membeli Instans Terpesan berikut di akun A:

- 1 x `m3.2xlarge` Amazon Linux, Instans Terpesan penghunian default di Wilayah us-east-1

Keuntungan Instans Terpesan diterapkan dengan cara berikut:

- Instans Terpesan regional `m3.2xlarge` di us-east-1 memberikan Zona Ketersediaan dan fleksibilitas ukuran instans, karena ini adalah Instans Terpesan Amazon Linux regional dengan penghunian default. Ini berlaku pertama untuk instans `m3.large`, kemudian ke instans `m3.xlarge`, karena berlaku dari ukuran instans terkecil hingga terbesar dalam keluarga instans berdasarkan faktor normalisasi.

  Sebuah `m3.large` contoh setara dengan 4 yang dinormalisasiunits/hour.

  Sebuah `m3.xlarge` contoh setara dengan 8 yang dinormalisasiunits/hour.

  Sebuah `m3.2xlarge` contoh setara dengan 16 yang dinormalisasiunits/hour.

  Manfaatnya diterapkan sebagai berikut:

  Instans `m3.2xlarge` Cadangan regional memberikan manfaat penuh untuk `m3.large` penggunaan 2x, karena bersama-sama instans ini menyumbang 8 yang dinormalisasi units/hour. Ini menyisakan 8 dinormalisasi units/hour untuk diterapkan pada `m3.xlarge` contoh.

  Dengan 8 yang tersisa dinormalisasi units/hour, Instans Cadangan `m3.2xlarge` regional memberikan manfaat penuh untuk `m3.xlarge` penggunaan 1 x, karena setiap `m3.xlarge` instans setara dengan 8 yang dinormalisasi units/hour. Sisa pem `m3.xlarge` akaian dibebankan sesuai On-Demand tarif.

Skenario 3: Instans Terpesan Regional dalam akun tertaut

Instans Terpesan pertama kali diterapkan untuk penggunaan dalam akun pembelian, diikuti dengan penggunaan yang memenuhi syarat di akun lain mana pun dalam organisasi. Untuk informasi selengkapnya, lihat Instans Terpesan dan penagihan gabungan. Untuk Instans Terpesan regional yang menawarkan fleksibilitas ukuran instans, keuntungan diterapkan dari ukuran instans terkecil hingga terbesar dalam keluarga instans tersebut.

Anda menjalankan On-Demand Instans berikut di akun A (akun pembelian):

- 2 x `m4.xlarge` Linux, instans penghunian default di Zona Ketersediaan us-east-1a
- 1 x `m4.2xlarge` Linux, instans penghunian default di Zona Ketersediaan us-east-1b
- 2 x `c4.xlarge` Linux, instans penghunian default di Zona Ketersediaan us-east-1a
- 1 x `c4.2xlarge` Linux, instans penghunian default di Zona Ketersediaan us-east-1b

Pelanggan lain menjalankan On-Demand Instans berikut di akun B—akun tertaut:

- 2 x `m4.xlarge` Linux, instans penghunian default di Zona Ketersediaan us-east-1a

Anda membeli Instans Terpesan wilayah berikut di akun A:

- 4 x `m4.xlarge` Linux, Instans Terpesan penghunian default di Wilayah us-east-1
- 2 x `c4.xlarge` Linux, Instans Terpesan penghunian default di Wilayah us-east-1

Keuntungan Instans Terpesan regional diterapkan dengan cara berikut:

- Diskon keempat Instans Terpesan `m4.xlarge` digunakan oleh kedua instans `m4.xlarge` dan satu instans `m4.2xlarge` di akun A (akun pembelian). Ketiga instans cocok dengan atribut (keluarga instans, Wilayah, platform, penghunian). Diskon diterapkan ke instans di akun pembelian (akun A) terlebih dahulu, meskipun akun B (akun tertaut) memiliki dua `m4.xlarge` yang juga cocok dengan Instans Terpesan. Tidak ada reservasi kapasitas karena Instans Terpesan adalah Instans Terpesan wilayah.
- Diskon dua Instans Terpesan `c4.xlarge` diterapkan ke kedua instans `c4.xlarge`, karena keduanya adalah ukuran instans yang lebih kecil daripada instans `c4.2xlarge`. Tidak ada reservasi kapasitas karena Instans Terpesan adalah Instans Terpesan wilayah.

Skenario 4: Instans Terpesan Zonal dalam akun tertaut

Secara umum, Instans Terpesan yang dimiliki oleh sebuah akun diterapkan terlebih dahulu ke penggunaan di akun tersebut. Namun, jika ada Instans Terpesan untuk Zona Ketersediaan tertentu (Instans Terpesan zonal) yang berkualifikasi dan tidak digunakan di akun lain dalam organisasi, instans tersebut diterapkan ke akun sebelum Instans Terpesan regional yang dimiliki oleh akun tersebut. Hal ini dilakukan untuk memastikan pemanfaatan Instans Terpesan yang maksimal dan tagihan yang lebih rendah. Untuk tujuan penagihan, semua akun di organisasi diperlakukan sebagai satu akun. Contoh berikut dapat membantu menjelaskan hal ini.

Anda menjalankan Inst On-Demand ans berikut di akun A (akun pembelian):

- 1 x `m4.xlarge` Linux, instans penghunian default di Zona Ketersediaan us-east-1a

Pelanggan menjalankan Inst On-Demand ans berikut di akun tertaut B:

- 1 x `m4.xlarge` Linux, instans penghunian default di Zona Ketersediaan us-east-1b

Anda membeli Instans Terpesan wilayah berikut di akun A:

- 1 x `m4.xlarge` Linux, Instans Terpesan penghunian default di Wilayah us-east-1

Seorang pelanggan juga membeli Instans Terpesan zonal berikut di akun C tertaut:

- 1 x `m4.xlarge` Linux, Instans Terpesan penghunian default di Zona Ketersediaan us-east-1a

Keuntungan Instans Terpesan diterapkan dengan cara berikut:

- Diskon dari Instans Terpesan zonal `m4.xlarge` yang dimiliki oleh akun C diterapkan ke penggunaan `m4.xlarge` di akun A.
- Diskon dari Instans Terpesan regional `m4.xlarge` yang dimiliki oleh akun A diterapkan ke penggunaan `m4.xlarge` di akun B.
- Jika Instans Cadangan regional yang dimiliki oleh akun A pertama kali diterapkan pada penggunaan di akun A, Instans Cadangan zona yang dimiliki oleh akun C tetap tidak digunakan dan penggunaan di akun B dikenakan On-Demand tarif.

Untuk informasi selengkapnya, lihat Memahami reservasi Anda di AWS Cost and Usage Report.

> **Note**
>
> Instans Terpesan Zona mereservasi kapasitas untuk akun pemilik saja dan tidak dapat dibagikan dengan Akun AWS lain. Jika Anda perlu berbagi kapasitas dengan Akun AWS lain, gunakan Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand.

## Menggunakan Instans Terpesan Anda

Instans Cadangan secara otomatis diterapkan ke On-Demand Instans yang sedang berjalan asalkan spesifikasinya cocok. Jika Anda tidak menjalankan On-Demand Instans yang sesuai dengan spesifikasi Instans Cadangan, Instans Cadangan tidak digunakan sampai Anda meluncurkan instans dengan spesifikasi yang diperlukan.

Jika Anda meluncurkan On-Demand Instans untuk memanfaatkan manfaat penagihan Instans Cadangan, pastikan Anda menentukan informasi berikut saat mengonfigurasi On-Demand Instans:

Platform

Anda harus menentukan Amazon Machine Image (AMI) yang cocok dengan platform (deskripsi produk) dari Instans Terpesan Anda. Misalnya, jika Anda menentukan `Linux/UNIX` untuk instans Terpesan, Anda dapat meluncurkan instans dari AMI Amazon Linux atau AMI Ubuntu.

Jenis instans

Jika membeli Instans Terpesan zonal, Anda harus menentukan tipe instans yang sama dengan Instans Terpesan Anda. Misalnya, `t3.large`. Untuk informasi selengkapnya, lihat Bagaimana Instans Terpesan zonal diterapkan.

Jika membeli Instans Terpesan regional, Anda harus menentukan tipe instans dari keluarga instans yang sama dengan tipe instans dari Instans Terpesan Anda. Misalnya, jika Anda menetapkan `t3.xlarge` untuk Instans Terpesan, Anda harus meluncurkan instans Anda dari keluarga T3, tetapi Anda dapat menentukan berapa pun ukuran apa pun. Misalnya, `t3.medium`. Untuk informasi selengkapnya, lihat Bagaimana Instans Terpesan regional diterapkan.

Zona Ketersediaan

Jika Anda membeli Instans Terpesan zonal untuk Zona Ketersediaan tertentu, Anda harus meluncurkan instans tersebut ke dalam Zona Ketersediaan yang sama.

Jika Anda membeli Instans Terpesan regional, Anda dapat meluncurkan instans ke Zona Ketersediaan mana pun di Wilayah yang Anda tentukan untuk Instans Terpesan tersebut.

Penghunian

Penghunian (`dedicated` atau `shared`) instans Anda harus cocok dengan penghunian Instans Terpesan. Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2.

Untuk contoh bagaimana Instans Cadangan diterapkan ke On-Demand Instans yang sedang berjalan, lihatBagaimana diskon Instans Cadangan diterapkan. Untuk informasi selengkapnya, lihat Mengapa Instans Cadangan Amazon EC2 saya tidak berlaku untuk AWS penagihan saya dengan cara yang saya harapkan?

Anda dapat menggunakan berbagai metode untuk meluncurkan On-Demand Instans yang menggunakan diskon Instans Cadangan Anda. Untuk informasi selengkapnya tentang berbagai

metode peluncuran, lihat Luncurkan instans Amazon EC2. Anda juga dapat menggunakan Amazon EC2 Auto Scaling untuk meluncurkan instans. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon EC2 Auto Scaling.

## Cara kerja penagihan dengan Instans Cadangan

Semua Instans Cadangan memberi Anda diskon dibandingkan dengan On-Demand harga. Dengan Instans Terpesan, Anda membayar untuk seluruh jangka waktu terlepas dari penggunaan sebenarnya. Anda dapat memilih untuk membayar Instans Terpesan di muka, sebagian di muka, atau bulanan, tergantung opsi pembayaran yang ditentukan untuk Instans Terpesan.

Saat Instans Cadangan kedaluwarsa, Anda akan dikenakan On-Demand tarif untuk penggunaan instans EC2. Anda dapat mengantrekan Instans Terpesan untuk pembelian hingga tiga tahun sebelumnya. Hal ini dapat membantu Anda memastikan bahwa Anda memiliki cakupan tanpa gangguan. Untuk informasi selengkapnya, lihat Mengantrekan pembelian Anda.

Ter AWS Tingkat Gratis sedia untuk yang baru Akun AWS. Jika Anda menggunakan AWS Tingkat Gratis untuk menjalankan instans Amazon EC2, dan membeli Instans Cadangan, Anda akan dikenakan harga standar. Untuk informasi, lihat AWS Tingkat Gratis.

Daftar Isi


- Penagihan penggunaan
- Melihat tagihan Anda
- Instans Terpesan dan penagihan gabungan
- Tingkat harga diskon Instans Terpesan


### Penagihan penggunaan

Instans Terpesan ditagih untuk setiap jam aktual selama jangka waktu yang Anda pilih, terlepas dari apakah sebuah instans sedang berjalan. Setiap jam aktual dimulai pada jam (nol menit dan nol detik setelah jam) aktual 24 jam standar. Misalnya, 1:00:00 hingga 1:59:59 adalah satu jam-jam. Untuk informasi selengkapnya tentang status instans, lihat Perubahan status instans Amazon EC2.

Manfaat penagihan Instans Cadangan dapat diterapkan ke instans yang sedang berjalan berdasarkan basis per detik. Per-second penagihan tersedia untuk instans yang menggunakan distribusi Linux open-source, seperti Amazon Linux dan Ubuntu. Per-hour penagihan digunakan untuk distribusi Linux komersial, seperti Red Hat Enterprise Linux dan SUSE Linux Enterprise Server.

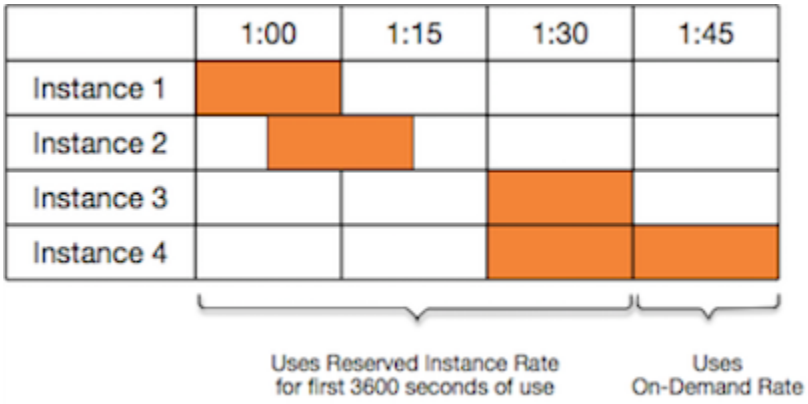
Keuntungan tagihan Instans Terpesan dapat diterapkan ke maksimum 3.600 detik (satu jam) penggunaan instans per jam aktual. Anda dapat menjalankan beberapa instans secara bersamaan, tetapi hanya dapat menerima manfaat diskon Instans Cadangan dengan total 3600 detik per jam-jam; penggunaan instans yang melebihi 3600 detik dalam satu jam akan ditagih dengan tarif tersebut. On-Demand

Misalnya, jika Anda membeli satu `m4.xlarge` Instans Cadangan dan menjalankan empat `m4.xlarge` instans secara bersamaan selama satu jam, satu instans dikenakan biaya satu jam penggunaan Instans Cadangan dan tiga instans lainnya dikenakan biaya On-Demand penggunaan tiga jam.

Namun, jika Anda membeli satu `m4.xlarge` Instans Cadangan dan menjalankan empat `m4.xlarge` instans selama 15 menit (900 detik) masing-masing dalam jam yang sama, total waktu berjalan untuk instans adalah satu jam, yang menghasilkan satu jam penggunaan Instans Cadangan dan 0 jam On-Demand penggunaan.




Jika beberapa instans yang memenuhi syarat berjalan secara bersamaan, manfaat penagihan Instans Cadangan diterapkan ke semua instans secara bersamaan hingga maksimum 3600 detik dalam satu jam; setelah itu, tarif berlaku. On-Demand




Penjelajah Biaya di konsol Penagihan dan Manajemen Biaya memungkinkan Anda menganalisis penghematan terhadap On-Demand Instans yang sedang berjalan. FAQ Instans Terpesan menyertakan contoh penghitungan nilai daftar.

Jika Anda menutup AWS akun, On-Demand penagihan untuk sumber daya Anda akan berhenti. Namun, jika Anda memiliki Instans Terpesan di akun, Anda terus menerima tagihan untuk ini hingga kedaluwarsa.

### Melihat tagihan Anda

Anda dapat mencari tahu tentang biaya dan tarif ke akun Anda dengan melihat konsol AWS Manajemen Penagihan dan Biaya tersebut.

- Dasbor menampilkan ringkasan pengeluaran untuk akun Anda.
- Pada halaman Tagihan, di bawah Detail, perluas bagian Elastic Compute Cloud dan Wilayah untuk mendapatkan informasi penagihan terkait Instans Terpesan Anda.

Anda dapat melihat tagihannya secara online, atau Anda dapat mengunduh file CSV.

Anda juga dapat melacak pemanfaatan Instans Cadangan menggunakan AWS Laporan Biaya dan Penggunaan. Untuk informasi selengkapnya, lihat Memahami reservasi Anda.

### Instans Terpesan dan penagihan gabungan

Keuntungan harga dari Instans Terpesan dibagikan ketika akun pembelian merupakan bagian dari sekumpulan akun yang ditagih dalam satu akun pembayar penagihan gabungan. Penggunaan instans di semua akun anggota dikumpulkan di akun pembayar setiap bulan. Hal ini biasanya berguna untuk perusahaan yang memiliki tim atau grup fungsional yang berbeda. Kemudian, logika Instans Terpesan normal diterapkan untuk menghitung tagihan. Untuk informasi selengkapnya, lihat Tagihan Gabungan untuk AWS Organizations.

Jika Anda menutup akun yang membeli Instans Terpesan, maka akun pembayar akan dikenai biaya untuk Instans Terpesan hingga Instans Terpesan tersebut kedaluwarsa. Setelah akun yang ditutup dihapus secara permanen dalam 90 hari, akun anggota tidak akan lagi mendapatkan keuntungan dari diskon penagihan Instans Terpesan.

> **Note**
>
> Instans Terpesan Zona mereservasi kapasitas untuk akun pemilik saja dan tidak dapat dibagikan dengan Akun AWS lain. Jika Anda perlu berbagi kapasitas dengan Akun AWS lain, gunakan Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand.

## Tingkat harga diskon Instans Terpesan

Jika memenuhi syarat untuk tingkat harga diskon, maka akun Anda secara otomatis menerima diskon di muka dan biaya penggunaan instans untuk pembelian Instans Terpesan yang Anda lakukan dalam level tingkat tersebut sejak saat itu. Agar memenuhi syarat untuk mendapatkan diskon, nilai daftar Instans Terpesan Anda di Wilayah harus sebanyak 500.000 USD atau lebih.

Aturan-aturan berikut berlaku:

- Tingkat harga dan diskon terkait hanya berlaku untuk pembelian Instans Terpesan Standar Amazon EC2.
- Tingkat harga tidak berlaku untuk Instans Terpesan Windows dengan SQL Server Standard, SQL Server Web, dan SQL Server Enterprise.
- Tingkat harga tidak berlaku untuk Instans Cadangan untuk Linux dengan SQL Server dalam bentuk apa pun.
- Diskon tingkat harga hanya berlaku untuk pembelian yang dilakukan dari AWS. Diskon ini tidak berlaku untuk pembelian Instans Terpesan pihak ketiga.
- Tingkat harga diskon saat ini tidak berlaku untuk pembelian Instans Terpesan Konvertibel.

Topik


- Menghitung diskon harga Instans Terpesan
- Membeli dengan tingkat diskon
- Melewati tingkat harga
- Penagihan gabungan untuk tingkatan harga


### Menghitung diskon harga Instans Terpesan

Anda dapat menentukan tingkat harga akun dengan menghitung nilai daftar untuk semua Instans Terpesan Anda di suatu Wilayah. Kalikan harga berulang per jam untuk setiap reservasi dengan total jumlah jam untuk jangka waktu tersebut dan tambahkan harga di muka yang tidak didiskon (juga dikenal sebagai harga tetap) pada saat pembelian. Karena didasarkan pada harga yang tidak didiskon (publik), nilai daftar tidak terpengaruh jika Anda memenuhi syarat untuk diskon volume atau jika harga turun setelah Anda membeli Instans Terpesan.

```
List value = fixed price + (undiscounted recurring hourly price * hours in term)
```

Misalnya, untuk Instans Terpesan `t2.small` Biaya di Muka Sebagian 1 tahun, asumsikan harga di muka adalah USD60,00 dan tarif per jam adalah USD0,007. Ini memberikan nilai daftar sebesar 121,32 USD.

```
121.32 = 60.00 + (0.007 * 8760)
```

Untuk melihat nilai harga tetap untuk Instans Cadangan menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans Terpesan.
3. Untuk menampilkan kolom Harga di muka, pilih pengaturan (  ) di pojok kanan atas, aktifkan Harga di muka, dan pilih Konfirmasi.

Untuk melihat nilai harga tetap untuk Instans Terpesan menggunakan baris perintah

- describe-reserved-instances (AWS CLI)
- Get-EC2ReservedInstance (AWS Tools for Windows PowerShell)

Membeli dengan tingkat diskon

Saat Anda membeli Instans Terpesan, Amazon EC2 secara otomatis menerapkan diskon ke bagian pembelian Anda yang termasuk dalam tingkat harga diskon. Anda tidak perlu melakukan sesuatu secara berbeda, dan Anda dapat membeli Instans Terpesan menggunakan salah satu alat Amazon EC2. Untuk informasi selengkapnya, lihat Beli Instans Cadangan untuk Amazon EC2.

Setelah nilai daftar Instans Terpesan aktif Anda di suatu Wilayah melintasi tingkat harga diskon, setiap pembelian Instans Terpesan di Wilayah tersebut pada masa mendatang akan dikenakan tarif diskon. Jika satu pembelian Instans Terpesan di suatu Wilayah membawa Anda melewati ambang batas tingkat diskon, maka porsi pembelian yang berada di atas ambang harga akan dikenakan tarif diskon. Untuk informasi selengkapnya tentang ID Instans Terpesan sementara yang dibuat selama proses pembelian, lihat Melewati tingkat harga.

Jika nilai daftar Anda berada di bawah titik harga untuk tingkat harga diskon tersebut—misalnya, jika beberapa Instans Terpesan Anda kedaluwarsa—pembelian Instans Terpesan di Wilayah ini pada masa mendatang tidak didiskon. Namun, Anda terus mendapatkan diskon yang berlaku pada setiap Instans Terpesan yang awalnya dibeli dalam tingkat harga diskon.

Saat Anda membeli Instans Terpesan, salah satu dari empat skenario mungkin terjadi:

- Tanpa diskon—Pembelian Anda dalam suatu Wilayah masih di bawah ambang batas diskon.
- Diskon sebagian—Pembelian Anda dalam suatu Wilayah melewati ambang batas tingkat diskon pertama. Tidak ada diskon yang diterapkan untuk satu atau lebih reservasi dan tarif diskon berlaku untuk reservasi yang tersisa.
- Diskon penuh—Seluruh pembelian Anda dalam suatu Wilayah termasuk dalam satu tingkat diskon dan didiskon dengan tepat.
- Dua tarif diskon—Pembelian Anda dalam suatu Wilayah melintasi dari tingkat diskon yang lebih rendah ke tingkat diskon yang lebih tinggi. Anda akan dikenai dua tarif berbeda: satu atau beberapa reservasi dengan tarif diskon lebih rendah, dan reservasi lainnya dengan tarif diskon lebih tinggi.

#### Melewati tingkat harga

Jika pembelian masuk ke tingkat harga diskon, Anda akan melihat banyak entri untuk pembelian itu: satu untuk bagian pembelian yang ditagih dengan harga reguler, dan yang lain untuk bagian pembelian yang ditagih dengan tarif diskon yang berlaku.

Layanan Instans Terpesan menghasilkan beberapa ID Instans Terpesan karena pembelian Anda lewat dari tingkat yang tidak didiskon, atau dari satu tingkat yang didiskon ke tingkat yang lain. Ada ID untuk setiap set reservasi dalam satu tingkatan. Akibatnya, ID yang dikembalikan oleh perintah CLI atau tindakan API pembelian Anda berbeda dari ID Instans Terpesan baru yang sebenarnya.

#### Penagihan gabungan untuk tingkatan harga

Akun penagihan gabungan menggabungkan nilai daftar akun anggota dalam satu Wilayah. Ketika nilai daftar dari semua Instans Terpesan yang aktif untuk akun penagihan gabungan mencapai tingkat harga diskon, setiap Instans Terpesan yang dibeli setelah titik ini oleh anggota mana pun dari akun penagihan gabungan akan dikenakan tarif diskon (selama nilai daftar untuk itu akun gabungan tersebut tetap di atas ambang batas tingkat harga diskon). Untuk informasi selengkapnya, lihat Instans Terpesan dan penagihan gabungan.

## Beli Instans Cadangan untuk Amazon EC2

Untuk membeli Instans Cadangan untuk Amazon EC2, Anda dapat menggunakan konsol Amazon EC2, alat baris perintah, atau SDK untuk mencari penawaran Instans Cadangan dari AWS dan penjual pihak ketiga, menyesuaikan parameter pencarian hingga Anda menemukan kecocokan yang tepat yang Anda cari.

Saat mencari Instans Terpesan untuk dibeli, Anda menerima penawaran kuota dari penawaran yang ditampilkan. Ketika Anda melanjutkan pembelian, AWS secara otomatis menempatkan harga batas pada harga pembelian. Total biaya Instans Terpesan Anda tidak akan melebihi jumlah kuota Anda.

Jika harga naik atau berubah untuk alasan apa pun, pembelian tidak selesai. Saat Anda membeli Instans Cadangan penjual pihak ketiga dari Marketplace Instans Cadangan Amazon EC2, jika ada penawaran yang mirip dengan pilihan Anda tetapi dengan harga di muka yang lebih rendah, Anda akan AWS menjual penawaran dengan harga awal yang lebih rendah.

Sebelum Anda mengonfirmasi pembelian, tinjau detail Instans Terpesan yang akan dibeli dan pastikan semua parameternya akurat. Setelah membeli Instans Cadangan (baik dari penjual pihak ketiga di Marketplace Instans Cadangan atau dari AWS), Anda tidak dapat membatalkan pembelian. Anda dapat mengantri pembelian untuk tanggal mendatang, dan membatalkan pembelian yang diantri sebelum waktu yang dijadwalkan.

Untuk membeli dan memodifikasi Instans Terpesan, pastikan bahwa pengguna Anda memiliki izin yang sesuai, seperti kemampuan untuk menjelaskan Zona Ktersediaan. Untuk informasi, lihat the section called “Cara Menggunakan Instans Cadangan” (API) atau the section called “Cara Menggunakan Instans Cadangan” (konsol).

Tugas

- Memilih platform
- Mengantrekan pembelian Anda
- Membeli Instans Terpesan Standar
- Membeli Instans Terpesan Konvertibel
- Membeli dari Marketplace Instans Terpesan
- Membatalkan antrean pembelian
- Memperbarui Instans Terpesan

## Memilih platform

Amazon EC2 mendukung platform berikut untuk Instans Cadangan:

- Linux/UNIX
- Linux dengan SQL Server Standard
- Linux dengan SQL Server Web

- Linux dengan SQL Server Enterprise
- SUSE Linux
- Linux Red Hat Enterprise
- Linux Red Hat Enterprise dengan HA
- Windows
- Windows dengan SQL Server Standard
- Windows dengan SQL Server Web
- Windows dengan SQL Server Enterprise

Pertimbangan-pertimbangan

- Jika Anda membawa langganan yang ada (BYOS) untuk Red Hat Enterprise Linux, SUSE Linux, atau Ubuntu Pro, Anda harus memilih penawaran untuk platform tersebut Linux/Unix.
- Instans Cadangan tidak didukung pada instans yang menjalankan macOS atau Ubuntu Pro (termasuk langganan EC2, yaitu, bukan BYOS). Untuk menghemat dengan harga On-Demand instans, sebaiknya gunakan instans macOS dan Ubuntu Pro (termasuk langganan EC2) dengan Paket Tabungan. Untuk informasi selengkapnya, lihat Panduan Pengguna Paket Tab ungan.

Untuk memastikan bahwa instance berjalan di Instans Cadangan tertentu, platform Instans Cadangan harus sesuai dengan platform AMI yang digunakan untuk meluncurkan instans. Untuk AMI Linux, penting untuk memeriksa apakah platform AMI menggunakan nilai umum Linux/UNIX atau nilai yang lebih spesifik seperti SUSE Linux.

Console

Untuk memeriksa platform AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pada tab Detail, catat nilai detail Platform.

AWS CLI

Untuk memeriksa platform AMI

Gunakan perintah describe-images dan periksa nilainya. `PlatformDetails`

```
aws ec2 describe-images \
    --image-id ami-0abcdef1234567890 \
    --query Images[*].PlatformDetails
```

Berikut ini adalah output contoh.

```
[
    "Linux/UNIX"
]
```

PowerShell

Untuk memeriksa platform AMI

Gunakan Get-EC2Image cmdlet dan periksa nilainya. `PlatformDetails`

```
Get-EC2Image `
    -ImageId ami-0abcdef1234567890 | `
    Select PlatformDetails
```

Berikut ini adalah output contoh.

```
PlatformDetails
---------------
Linux/UNIX
```

### Mengantrekan pembelian Anda

Secara default, saat Anda membeli Instans Terpesan, pembelian tersebut langsung dibuat. Atau, Anda dapat mengantrekan pembelian untuk tanggal dan waktu pada masa mendatang. Misalnya, Anda dapat mengantrekan pembelian sekitar waktu Instans Terpesan yang ada kedaluwarsa. Hal ini dapat membantu Anda memastikan bahwa Anda memiliki cakupan tanpa gangguan.

Anda dapat mengantrekan pembelian untuk Instans Terpesan regional, tetapi tidak untuk Instans Terpesan zonal atau Instans Terpesan dari penjual lain. Anda dapat mengantrekan pembelian hingga tiga tahun ke depan. Pada tanggal dan waktu yang dijadwalkan, pembelian dilakukan menggunakan metode pembayaran default. Setelah pembayaran berhasil, keuntungan penagihan diterapkan.

Anda dapat menetapkan tanggal untuk pembelian yang antri di konsol Amazon EC2, dan pembelian akan mengantri hingga 00:00 UTC pada tanggal ini. Untuk menentukan waktu yang berbeda untuk pembelian yang mengantri, gunakan AWS SDK atau alat baris perintah.

Anda dapat melihat antrean pembelian Anda di konsol Amazon EC2. Status antrean pembelian adalah antre. Anda dapat membatalkan antrean pembelian kapan saja sebelum waktu yang dijadwalkan. Untuk detailnya, lihat Membatalkan antrean pembelian.

Membeli Instans Terpesan Standar

Anda dapat membeli Instans Terpesan Standar di Zona Ketersediaan tertentu dan mendapatkan reservasi kapasitas. Atau, Anda dapat melepaskan reservasi kapasitas dan membeli Instans Terpesan Standar regional.

Setelah pembelian selesai, jika Anda sudah memiliki instans berjalan yang cocok dengan spesifikasi Instans Terpesan, keuntungan penagihan langsung diterapkan. Anda tidak perlu me-restart instans Anda. Jika Anda tidak memiliki instans berjalan yang sesuai, luncurkan instance dan pastikan Anda cocok dengan kriteria yang tepat yang Anda tentukan untuk Instans Cadangan. Untuk informasi selengkapnya, lihat Menggunakan Instans Terpesan Anda.

Untuk contoh tentang bagaimana Instans Terpesan diterapkan ke instans berjalan Anda, lihat Bagaimana diskon Instans Cadangan diterapkan.

Console

Untuk membeli Instans Cadangan Standar

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans Terpesan, lalu pilih Beli Instans Terpesan.
3. Untuk Kelas Penawaran, pilih Standar untuk menampilkan Tnstans Terpesan Standar.
4. Untuk membeli reservasi kapasitas, aktifkan Hanya tampilkan penawaran yang memiliki kapasitas di sudut kanan atas layar pembelian. Ketika Anda mengaktifkan pengaturan ini, bidang Zona Ketersediaan akan muncul.

   Untuk membeli Instans Terpesan wilayah, nonaktifkan pengaturan ini. Ketika Anda menonaktifkan pengaturan ini, bidang Zona Ketersediaan akan hilang.
5. Pilih konfigurasi lain sesuai kebutuhan, lalu pilih Cari.
6. Untuk setiap Instans Terpesan yang ingin Anda beli, masukkan jumlah yang diinginkan, dan pilih Tambahkan ke keranjang.

Untuk membeli Instans Terpesan Standar dari Marketplace Instans Terpesan, cari Pihak ke-3 di kolom Penjual pada hasil pencarian. Kolom Istilah menampilkan istilah nonstandar. Untuk informasi selengkapnya, lihat Membeli dari Marketplace Instans Terpesan.

7. Untuk melihat ringkasan Instans Terpesan yang Anda pilih, klik Lihat keranjang.
8. Jika Pesanan pada adalah Sekarang, pembelian akan segera diselesaikan setelah Anda memilih Pesan semua. Untuk mengantrekan pembelian, pilih Sekarang dan pilih tanggal. Anda dapat memilih tanggal yang berbeda untuk setiap penawaran yang memenuhi syarat di keranjang. Pembelian diantrekan sampai pukul 00:00 UTC pada tanggal yang dipilih.
9. Untuk menyelesaikan pesanan, pilih Pesan semua.

   Jika, pada saat melakukan pemesanan, ada penawaran yang mirip dengan pilihan Anda tetapi dengan harga yang lebih rendah, AWS menjual penawaran dengan harga lebih rendah.
10. Pilih Tutup.

    Status pesanan Anda tercantum di kolom Status. Ketika pesanan Anda selesai, nilai Status berubah dari `Payment-pending` menjadi.`Active` Ketika Instans Terpesan sudah `Active`, instans tersebut siap digunakan.

    Jika statusnya masuk ke`Retired`, AWS mungkin belum menerima pembayaran Anda.

AWS CLI

Untuk membeli Instans Cadangan Standar

1. Temukan Instans Terpesan yang tersedia menggunakan perintah describe-reserved-instances-offerings. Tentukan `--offering-class` opsi `standard` untuk mengembalikan Instans Cadangan Standar saja. Anda dapat menerapkan kriteria tambahan untuk mempersempit hasil Anda. Misalnya, gunakan perintah berikut untuk membeli Instans Cad `t2.large` angan regional dengan sewa default `Linux/UNIX` untuk jangka waktu 1 tahun saja.

```
aws ec2 describe-reserved-instances-offerings \
    --instance-type t2.large \
    --offering-class standard \
    --product-description "Linux/UNIX" \
    --instance-tenancy default \
    --filters Name=duration,Values=31536000 \
```

```
    Name=scope,Values=Region
```

Untuk menemukan Instans Terpesan di Marketplace Instans Terpesan saja, gunakan filter `marketplace` dan jangan tentukan durasi dalam permintaan karena jangka waktu mungkin lebih pendek dari jangka waktu 1 atau 3 tahun.

```
aws ec2 describe-reserved-instances-offerings \
    --instance-type t2.large \
    --offering-class standard \
    --product-description "Linux/UNIX" \
    --instance-tenancy default \
    --filters Name=marketplace,Values=true
```

Saat Anda menemukan Instans Terpesan yang memenuhi kebutuhan Anda, catat ID penawarannya. Misalnya:

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

2. Gunakan perintah purchase-reserved-instances-offering untuk membeli Instans Terpesan Anda. Anda harus menentukan ID penawaran Instans Terpesan yang Anda peroleh pada langkah sebelumnya dan Anda harus menentukan jumlah instans untuk reservasi.

```
aws ec2 purchase-reserved-instances-offering \
    --reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
    --instance-count 1
```

Secara default, pembelian segera diselesaikan. Atau, untuk mengantri pembelian, tambahkan opsi berikut ke panggilan sebelumnya.

```
--purchase-time "2020-12-01T00:00:00Z"
```

3. Gunakan perintah describe-reserved-instanss untuk mendapatkan status Instans Cadangan Anda.

```
aws ec2 describe-reserved-instances \
    --reserved-instances-ids b847fa93-e282-4f55-b59a-1342fec06327 \
    --query ReservedInstances[].State
```

PowerShell

Untuk membeli Instans Cadangan Standar

1. Temukan Instans Cadangan yang tersedia menggunakan Get-EC2ReservedInstancesOffering cmdlet. Tetapkan `standard` untuk parameter `-OfferingClass` agar hanya menampilkan Instans Terpesan Standar. Anda dapat menerapkan kriteria tambahan untuk mempersempit hasil Anda. Misalnya, gunakan perintah berikut untuk membeli Instans Cad `t2.large` angan regional dengan sewa default `Linux/UNIX` untuk jangka waktu 1 tahun saja.

```
Get-EC2ReservedInstancesOffering `
    -InstanceType "t2.large" `
    -OfferingClass "standard" `
    -ProductDescription "Linux/UNIX" `
    -InstanceTenancy "default" `
    -Filters @{Name="duration"; Values="31536000"} `
             @{Name="scope"; Values="Region"
```

   Untuk menemukan Instans Terpesan di Marketplace Instans Terpesan saja, gunakan filter `marketplace` dan jangan tentukan durasi dalam permintaan karena jangka waktu mungkin lebih pendek dari jangka waktu 1 atau 3 tahun.

```
Get-EC2ReservedInstancesOffering `
    -InstanceType t2.large `
    -OfferingClass "standard" `
    -ProductDescription "Linux/UNIX" `
    -InstanceTenancy default `
    -Filters @{Name="marketplace"; Values="true"}
```

   Saat Anda menemukan Instans Terpesan yang memenuhi kebutuhan Anda, catat ID penawarannya. Contoh:

```
bec624df-a8cc-4aad-a72f-4f8abc34caf2
```

2. Gunakan New-EC2ReservedInstance cmdlet untuk membeli Instans Cadangan Anda. Anda harus menentukan ID penawaran Instans Terpesan yang Anda peroleh pada langkah sebelumnya dan Anda harus menentukan jumlah instans untuk reservasi.

```
New-EC2ReservedInstance `
```

```
    -ReservedInstancesOfferingId "bec624df-a8cc-4aad-a72f-4f8abc34caf2" `
    -InstanceCount 1
```

Secara default, pembelian segera diselesaikan. Atau, untuk mengantrekan pembelian, tambahkan parameter berikut ke panggilan sebelumnya.

```
-PurchaseTime "2020-12-01T00:00:00Z"
```

3. Gunakan Get-EC2ReservedInstance cmdlet untuk mendapatkan status Instans Cadangan Anda.

```
Get-EC2ReservedInstance `
    -ReservedInstancesId b847fa93-e282-4f55-b59a-1342fec06327 | `
    Select State
```

Membeli Instans Terpesan Konvertibel

Anda dapat membeli Instans Terpesan Konvertibel di Zona Ketersediaan tertentu dan mendapatkan reservasi kapasitas. Atau, Anda dapat melepaskan reservasi kapasitas dan membeli Instans Terpesan Konvertibel regional.

Jika Anda sudah memiliki instans berjalan yang cocok dengan spesifikasi Instans Terpesan, keuntungan penagihan langsung diterapkan. Anda tidak perlu memulai ulang instans Anda. Jika Anda tidak memiliki instans berjalan yang cocok, luncurkan sebuah instans dan pastikan kesamaannya dengan kriteria yang sudah Anda tentukan untuk Instans Terpesan Anda. Untuk informasi selengkapnya, lihat Menggunakan Instans Terpesan Anda.

Untuk contoh tentang bagaimana Instans Terpesan diterapkan ke instans berjalan Anda, lihat Bagaimana diskon Instans Cadangan diterapkan.

Console

Untuk membeli Instans Cadangan yang Dapat Dikonversi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans Terpesan, lalu pilih Beli Instans Terpesan.
3. Untuk Kelas Penawaran, pilih Konvertibel untuk menampilkan Instans Terpesan Konvertibel.

4. Untuk membeli reservasi kapasitas, aktifkan Hanya tampilkan penawaran yang memiliki kapasitas di sudut kanan atas layar pembelian. Ketika Anda mengaktifkan pengaturan ini, bidang Zona Ketersediaan akan muncul.

   Untuk membeli Instans Terpesan wilayah, nonaktifkan pengaturan ini. Ketika Anda menonaktifkan pengaturan ini, bidang Zona Ketersediaan akan hilang.
5. Pilih konfigurasi lain sesuai kebutuhan dan pilih Cari.
6. Untuk setiap Instans Terpesan Konvertibel yang ingin Anda beli, masukkan jumlahnya, dan pilih Tambahkan ke keranjang.
7. Untuk melihat ringkasan pilihan Anda, pilih Lihat keranjang.
8. Jika Pesanan pada adalah Sekarang, pembelian akan segera diselesaikan setelah Anda memilih Pesan semua. Untuk mengantrekan pembelian, pilih Sekarang dan pilih tanggal. Anda dapat memilih tanggal yang berbeda untuk setiap penawaran yang memenuhi syarat di keranjang. Pembelian diantrekan sampai pukul 00:00 UTC pada tanggal yang dipilih.
9. Untuk menyelesaikan pesanan, pilih Pesan semua.

   Jika, pada saat melakukan pemesanan, ada penawaran yang mirip dengan pilihan Anda tetapi dengan harga yang lebih rendah, AWS menjual penawaran dengan harga lebih rendah.
10. Pilih Tutup.

    Status pesanan Anda tercantum di kolom Status. Ketika pesanan Anda selesai, nilai Status berubah dari `Payment-pending` menjadi.`Active` Ketika Instans Terpesan sudah `Active`, instans tersebut siap digunakan.

    Jika statusnya masuk ke`Retired`, AWS mungkin belum menerima pembayaran Anda.

AWS CLI

Cara membeli Instans Cadangan yang Dapat Dikonversi

1. Temukan Instans Terpesan yang tersedia menggunakan perintah describe-reserved-instances-offerings. Tentukan `--offering-class` opsi `convertible` untuk mengembalikan hanya Instans Cadangan yang Dapat Dikonversi. Anda dapat menerapkan kriteria tambahan untuk mempersempit hasil Anda. Misalnya, gunakan perintah berikut untuk membeli Instans Cad `t2.large` angan regional dengan sewa default untuk`Linux/UNIX`.

   ```
   aws ec2 describe-reserved-instances-offerings \
   ```

```
    --instance-type t2.large \
    --offering-class convertible \
    --product-description "Linux/UNIX" \
    --instance-tenancy default \
    --filters Name=scope,Values=Region
```

Saat Anda menemukan Instans Terpesan yang memenuhi kebutuhan Anda, catat ID penawarannya. Misalnya:

```
"ReservedInstancesOfferingId": "bec624df-a8cc-4aad-a72f-4f8abc34caf2"
```

2. Gunakan perintah purchase-reserved-instances-offering untuk membeli Instans Terpesan Anda. Anda harus menentukan ID penawaran Instans Terpesan yang Anda peroleh pada langkah sebelumnya dan Anda harus menentukan jumlah instans untuk reservasi.

```
aws ec2 purchase-reserved-instances-offering \
    --reserved-instances-offering-id bec624df-a8cc-4aad-a72f-4f8abc34caf2 \
    --instance-count 1
```

Secara default, pembelian segera diselesaikan. Atau, untuk mengantri pembelian, tambahkan opsi berikut ke panggilan sebelumnya.

```
--purchase-time "2020-12-01T00:00:00Z"
```

3. Gunakan perintah describe-reserved-instanss untuk mendapatkan status Instans Cadangan Anda.

```
aws ec2 describe-reserved-instances \
    --reserved-instances-ids b847fa93-e282-4f55-b59a-1342fec06327 \
    --query ReservedInstances[].State
```

PowerShell

Cara membeli Instans Cadangan yang Dapat Dikonversi

1. Temukan Instans Cadangan yang tersedia menggunakan Get-EC2ReservedInstancesOffering cmdlet. Tentukan `convertible` untuk parameter `-OfferingClass` agar hanya menampilkan Instans Terpesan Konvertibel. Anda dapat menerapkan kriteria tambahan untuk mempersempit hasil Anda. Misalnya, gunakan

perintah berikut untuk membeli Instans Cad `t2.large` angan regional dengan sewa default untuk`Linux/UNIX`.

```
Get-EC2ReservedInstancesOffering `
    -InstanceType "t2.large" `
    -OfferingClass "convertible" `
    -ProductDescription "Linux/UNIX" `
    -InstanceTenancy "default" `
    -Filters @{Name="scope"; Values="Region"
```

Saat Anda menemukan Instans Terpesan yang memenuhi kebutuhan Anda, catat ID penawarannya. Contoh:

```
bec624df-a8cc-4aad-a72f-4f8abc34caf2
```

2. Gunakan New-EC2ReservedInstance cmdlet untuk membeli Instans Cadangan Anda. Anda harus menentukan ID penawaran Instans Cadangan yang Anda peroleh pada langkah sebelumnya dan Anda harus menentukan jumlah instans untuk reservasi.

```
New-EC2ReservedInstance `
    -ReservedInstancesOfferingId "bec624df-a8cc-4aad-a72f-4f8abc34caf2" `
    -InstanceCount 1
```

Secara default, pembelian segera diselesaikan. Atau, untuk mengantrekan pembelian, tambahkan parameter berikut ke panggilan sebelumnya.

```
-PurchaseTime "2020-12-01T00:00:00Z"
```

3. Gunakan Get-EC2ReservedInstance cmdlet untuk mendapatkan status Instans Cadangan Anda.

```
Get-EC2ReservedInstance `
    -ReservedInstancesId b847fa93-e282-4f55-b59a-1342fec06327 | `
    Select State
```

### Membeli dari Marketplace Instans Terpesan

Anda dapat membeli Instans Terpesan dari penjual pihak ketiga yang memiliki Instans Terpesan yang tidak lagi diperlukan dari Marketplace Instans Terpesan. Anda dapat melakukan hal ini menggunakan

konsol Amazon EC2 atau alat baris perintah. Prosesnya mirip dengan membeli Instans Cadangan dari AWS. Untuk informasi selengkapnya, lihat Membeli Instans Terpesan Standar.

Ada beberapa perbedaan antara Instans Cadangan yang dibeli di Marketplace Instans Cadangan dan Instans Cadangan yang dibeli langsung dari AWS:

- Jangka Waktu – Instans Terpesan yang Anda beli dari penjual pihak ketiga memiliki sisa jangka waktu kurang dari standar penuh. Istilah standar penuh mulai AWS berjalan selama satu tahun atau tiga tahun.
- Harga di muka — Inst Third-party ans Cadangan dapat dijual dengan harga dimuka yang berbeda. Biaya penggunaan atau berulang tetap sama dengan biaya yang ditetapkan saat Instans Terpesan awalnya dibeli dari AWS.
- Tipe Instans Terpesan – Hanya Instans Terpesan Standar Amazon EC2 yang dapat dibeli dari Marketplace Instans Terpesan. Instans Cadangan yang Dapat Dikonversi, Amazon RDS, dan ElastiCache Instans Cadangan Amazon tidak tersedia untuk dibeli di Marketplace Instans Cadangan.

Informasi dasar tentang Anda dibagikan dengan penjual. Misalnya, kode pos dan informasi negara Anda.

Informasi ini memungkinkan penjual untuk menghitung pajak transaksi yang diperlukan yang harus mereka serahkan kepada pemerintah (seperti pajak penjualan atau pajak pertambahan nilai) dan disediakan sebagai laporan pencairan. Dalam keadaan yang jarang terjadi, AWS mungkin harus memberikan alamat email Anda kepada penjual, sehingga mereka dapat menghubungi Anda mengenai pertanyaan yang terkait dengan penjualan (misalnya, pertanyaan pajak).

Untuk alasan yang AWS sama, bagikan nama badan hukum penjual pada faktur pembelian pembeli. Jika Anda memerlukan informasi tambahan tentang penjual untuk pajak atau alasan terkait, hubungi Dukungan.

Membatalkan antrean pembelian

Anda dapat mengantrekan pembelian hingga tiga tahun ke depan. Anda dapat membatalkan antrean pembelian kapan saja sebelum waktu yang dijadwalkan.

Console

Untuk membatalkan pembelian antrian

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans Terpesan.
3. Pilih satu atau lebih Instans Terpesan.
4. Pilih Tindakan, Hapus antrean Instans Terpesan.
5. Ketika diminta untuk mengonfirmasi, masukkan Hapus, lalu Tutup.

AWS CLI

Untuk membatalkan antrean pembelian

Gunakan perintah delete-queued-reserved-instance.

```
aws ec2 delete-queued-reserved-instances \
    --reserved-instances-ids b847fa93-e282-4f55-b59a-1342fec06327
```

PowerShell

Untuk membatalkan antrean pembelian

Gunakan Remove-EC2QueuedReservedInstance cmdlet.

```
Remove-EC2QueuedReservedInstance `
    -ReservedInstancesId b847fa93-e282-4f55-b59a-1342fec06327
```

## Memperbarui Instans Terpesan

Anda dapat memperbarui Instans Terpesan sebelum dijadwalkan kedaluwarsa. Memperbarui Instans Terpesan akan mengantrekan pembelian Instans Terpesan dengan konfigurasi yang sama hingga Instans Terpesan saat ini kedaluwarsa.

Anda harus memperbarui Instans Cadangan menggunakan konsol Amazon EC2.

Untuk memperpanjang Instans Terpesan menggunakan pembelian yang diantrekan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans Terpesan.
3. Pilih Instans Terpesan yang akan diperpanjang.
4. Pilih Tindakan, Perpanjang Instans Terpesan.
5. Untuk menyelesaikan pesanan, pilih Pesan semua, lalu Tutup.

## Jual Instans Cadangan untuk Amazon EC2 di Marketplace Instans Cadangan

Amazon EC2 Reserved Instance Marketplace adalah platform yang memfasilitasi penjualan Instans Cadangan Standar yang tidak digunakan dari AWS pelanggan dan penjual pihak ketiga. Instans Cadangan ini dapat bervariasi dalam jangka waktu dan opsi harga. Anda mungkin ingin menjual Instans Cadangan saat Anda tidak lagi membutuhkannya, misalnya saat memindahkan instans ke instans baru Wilayah AWS, mengubah ke jenis instans yang berbeda, menyelesaikan proyek sebelum jangka waktu Instans Cadangan berakhir, kebutuhan bisnis Anda berubah, atau Anda memiliki kelebihan kapasitas.

Segera setelah Anda mencantumkan Instans Terpesan di Marketplace Instans Terpesan, instans tersebut akan tersedia untuk ditemukan oleh calon pembeli. Semua Instans Terpesan dikelompokkan menurut durasi sisa jangka waktu dan harga per jam.

Untuk memenuhi permintaan pembeli untuk membeli Instans Cadangan penjual pihak ketiga melalui Marketplace Instans Cadangan, AWS pertama-tama jual Instans Cadangan dengan harga dimuka terendah dalam pengelompokan yang ditentukan. Kemudian, AWS jual Instans Cadangan dengan harga terendah berikutnya, hingga seluruh pesanan pembeli terpenuhi. AWS kemudian memproses transaksi dan mentransfer kepemilikan Instans Cadangan kepada pembeli.

Anda memiliki Instans Terpesan hingga terjual. Setelah penjualan, Anda telah melepaskan reservasi kapasitas dan diskon biaya berulang. Jika Anda terus menggunakan instans, Anda akan dikenakan AWS On-Demand biaya mulai dari saat Instans Cadangan Anda dijual.

Jika ingin menjual Instans Terpesan yang tidak digunakan di Marketplace Instans Terpesan, Anda harus memenuhi kriteria kelayakan tertentu.

Untuk informasi tentang membeli Instans Terpesan di Marketplace Instans Terpesan, lihat Membeli dari Marketplace Instans Terpesan.

Daftar Isi


- Pembatasan dan batasan
- Mendaftar sebagai penjual

- Rekening bank untuk pencairan
- Informasi pajak
- Menentukan Harga Instans Terpesan Anda
- Mengiklankan Instans Terpesan Anda
- Status iklan Instans Terpesan
- Siklus hidup iklan
- Setelah Instans Terpesan Anda terjual
- Mendapatkan pembayaran
- Informasi yang dibagikan dengan pembeli

Pembatasan dan batasan

Sebelum dapat menjual reservasi yang tidak digunakan, Anda harus mendaftar sebagai penjual di Marketplace Instans Terpesan. Untuk informasi, lihat Mendaftar sebagai penjual.

Batasan dan larangan berikut berlaku saat menjual Instans Terpesan:

- Hanya Instans Terpesan regional dan zonal Standar Amazon EC2 yang dapat dijual di Marketplace Instans Terpesan.
- Instans Cadangan Konvertibel Amazon EC2 tidak dapat dijual di Marketplace Instans Cadangan.
- Instans Cadangan untuk AWS layanan lain, seperti Amazon RDS dan Amazon ElastiCache, tidak dapat dijual di Marketplace Instans Cadangan.
- Harus ada setidaknya satu bulan tersisa dalam jangka waktu Instans Terpesan Standar.
- Anda tidak dapat menjual Instans Cadangan Standar di Wilayah yang dinonaktifkan secara default.
- Harga minimum yang diizinkan di Marketplace Instans Terpesan adalah 0,00 USD.
- Anda dapat menjual Instans Terpesan Tanpa Biaya di Muka, Sebagian di Muka, atau Semua di Muka di Marketplace Instans Terpesan selama instans aktif di akun Anda setidaknya selama 30 hari. Selain itu, jika ada pembayaran di muka pada Instans Cadangan, itu hanya dapat dijual setelah AWS menerima pembayaran di muka.
- Anda tidak dapat menjual Instans Cadangan di Marketplace Instans Cadangan jika membelinya menggunakan diskon volume.
- Anda tidak dapat mengubah daftar di Marketplace Instans Terpesan secara langsung. Namun, Anda dapat mengubah daftar Anda dengan membatalkannya terlebih dahulu, lalu membuat daftar

lain dengan parameter baru. Untuk informasi, lihat Menentukan Harga Instans Terpesan Anda. Anda juga dapat memodifikasi Instans Terpesan sebelum mendaftarnya. Untuk informasi, lihat Memodifikasi Instans Terpesan.

- AWS membebankan biaya layanan sebesar 12 persen dari total harga di muka setiap Instans Cadangan Standar yang Anda jual di Marketplace Instans Cadangan. Harga di muka adalah harga yang dibebankan penjual untuk Instans Terpesan Standar.
- Saat Anda mendaftar sebagai penjual, bank yang Anda tentukan harus memiliki alamat AS. Untuk informasi selengkapnya, lihat Persyaratan penjual tambahan untuk produk berbayar di Panduan Penjual AWS Marketplace .
- Pelanggan Amazon Web Services India Private Limited (AWS India) tidak dapat mendaftar sebagai penjual di EC2 Reserved Instance Marketplace dan tidak dapat mencantumkan atau menjual Instans Cadangan di Pasar Instans Cadangan EC2, meskipun mereka memiliki rekening bank AS. Untuk informasi selengkapnya, lihat Apa perbedaan antara Akun AWS dan akun AWS India?
- Jika Anda mengubah penjual catatan menjadi Amazon Web Services India Private Limited (AWS India), Anda akan dibatalkan pendaftaran sebagai penjual dari Marketplace Instans Cadangan EC2 dan semua daftar aktif Anda yang ada di Pasar Instans Cadangan EC2 akan dihapus. Untuk mengembalikan status penjual Anda, Anda harus mengubah lokasi akun Anda ke negara selain India dan menyelesaikan proses pendaftaran penjual lagi.

### Mendaftar sebagai penjual

> **Note**
>
> Hanya yang Pengguna root akun AWS dapat mendaftarkan akun sebagai penjual.

Untuk menjual di Marketplace Instans Terpesan, Anda harus mendaftar sebagai penjual terlebih dahulu. Selama pendaftaran, Anda memberikan informasi berikut:

- Informasi bank —AWS harus memiliki informasi bank Anda untuk mencairkan dana yang dikumpulkan saat Anda menjual reservasi Anda. Bank yang Anda tentukan harus memiliki alamat AS. Untuk informasi selengkapnya, lihat Rekening bank untuk pencairan.
- Informasi pajak—Semua penjual wajib menyelesaikan wawancara informasi pajak untuk menentukan kewajiban pelaporan pajak yang diperlukan. Untuk informasi selengkapnya, lihat Informasi pajak.

Setelah AWS menerima pendaftaran penjual yang telah selesai, Anda menerima email yang mengonfirmasi pendaftaran Anda dan memberi tahu Anda bahwa Anda dapat mulai menjual di Marketplace Instans Cadangan.

Rekening bank untuk pencairan

AWS harus memiliki informasi bank Anda untuk mencairkan dana yang dikumpulkan saat Anda menjual Instans Cadangan. Bank yang Anda tentukan harus memiliki alamat di AS. Untuk informasi selengkapnya, lihat Persyaratan penjual tambahan untuk produk berbayar di Panduan Penjual AWS Marketplace .

Untuk mendaftarkan rekening bank default untuk pencairan

1. Buka halaman Pendaftaran Penjual Marketplace Instans Terpesan dan masuk menggunakan kredensial AWS Anda.
2. Pada halaman Kelola Rekening Bank, berikan informasi tentang bank berikut untuk menerima pembayaran:

   - Nama Pemilik Rekening Bank
   - Nomor perutean
   - Nomor rekening
   - Tipe rekening bank

> **Note**
>
> Jika Anda menggunakan rekening bank perusahaan, Anda diminta untuk mengirim informasi tentang rekening bank melalui faks (1-206-765-3424).

Setelah pendaftaran, rekening bank yang diberikan ditetapkan sebagai default, menunggu verifikasi dari bank. Diperlukan waktu hingga dua minggu untuk memverifikasi rekening bank baru, selama itu Anda tidak dapat menerima pencairan. Untuk rekening yang sudah ditetapkan, biasanya diperlukan waktu sekitar dua hari untuk menyelesaikan pembayaran.

Untuk mengubah rekening bank default untuk pencairan

1. Pada halaman Pendaftaran Penjual Marketplace Instans Terpesan, masuk dengan akun yang Anda gunakan saat mendaftar.

2. Pada halaman Kelola Rekening Bank, tambahkan rekening bank baru atau ubah rekening bank default sesuai kebutuhan.

Informasi pajak

Penjualan Instans Terpesan Anda mungkin dikenai pajak berbasis transaksi, seperti pajak penjualan atau pajak pertambahan nilai. Anda harus memeriksanya dengan departemen pajak, hukum, keuangan, atau akuntansi bisnis Anda untuk menentukan apakah pajak berbasis transaksi berlaku. Anda bertanggung jawab untuk mengumpulkan dan mengirim pajak berbasis transaksi ke otoritas pajak yang sesuai.

Sebagai bagian dari proses pendaftaran penjual, Anda harus menyelesaikan wawancara pajak di Portal Pendaftaran Penjual. Wawancara mengumpulkan informasi pajak Anda dan mengisi formulir IRS W-9, W-8BEN, atau W-8BEN-E, yang digunakan untuk menentukan kewajiban pelaporan pajak yang diperlukan.

Informasi pajak yang Anda masukkan sebagai bagian dari wawancara pajak mungkin berbeda, bergantung pada apakah Anda beroperasi sebagai individu atau bisnis, dan apakah Anda atau bisnis Anda adalah orang atau entitas AS atau non-AS. Saat Anda mengisi wawancara pajak, perhatikan hal-hal berikut:

- Informasi yang diberikan oleh AWS, termasuk informasi dalam topik ini, bukan merupakan saran pajak, hukum, atau profesional lainnya. Untuk mengetahui bagaimana persyaratan pelaporan IRS dapat memengaruhi bisnis Anda, atau jika Anda memiliki pertanyaan lain, hubungi penasihat pajak, hukum, atau profesional lainnya.
- Untuk memenuhi persyaratan pelaporan IRS seefisien mungkin, jawab semua pertanyaan dan masukkan semua informasi yang diminta selama wawancara.
- Periksa jawaban Anda. Hindari salah eja atau salah memasukkan nomor identifikasi pajak. Kesalahan tersebut dapat mengakibatkan formulir pajak tidak valid.

Berdasarkan respons wawancara pajak dan ambang batas pelaporan IRS Anda, Amazon mungkin mengajukan Formulir 1099-K. Amazon mengirimkan salinan Formulir 1099-K Anda pada atau sebelum tanggal 31 Januari pada tahun setelah tahun ketika akun pajak Anda mencapai tingkat ambang batas. Misalnya, jika akun Anda mencapai ambang batas pada tahun 2018, Formulir 1099-K Anda akan dikirimkan pada atau sebelum tanggal 31 Januari 2019.

Untuk informasi lebih lanjut tentang persyaratan IRS dan Formulir 1099-K, lihat FAQ Formu lir 1099-K di situs web IRS.

### Menentukan Harga Instans Terpesan Anda

Saat menetapkan harga untuk Instans Terpesan Anda, pertimbangkan hal berikut:

- Harga di muka – Harga di muka adalah satu-satunya harga yang dapat Anda tentukan untuk Instans Terpesan yang Anda jual. Harga dimuka adalah harga satu kali yang dibayar pembeli saat membeli setiap Instans Cadangan.

  Karena nilai Instans Cadangan menurun dari waktu ke waktu, secara default, harga AWS dapat diturunkan dengan kenaikan yang sama dari bulan ke bulan. Namun, Anda dapat menetapkan harga di muka yang berbeda berdasarkan kapan reservasi Anda terjual. Misalnya, jika Instans Terpesan Anda memiliki sisa jangka waktu sembilan bulan, Anda dapat menentukan jumlah yang ingin Anda terima jika pelanggan membeli Instans Terpesan tersebut dengan sembilan bulan tersisa. Anda dapat menetapkan harga lain dengan sisa lima bulan, dan harga lain dengan sisa satu bulan.

  Harga minimum yang diizinkan di Pasar Instans Terpesan adalah 0,00 USD.
- Batas – Batasan penjualan Instans Terpesan berikut berlaku untuk masa pakai Akun AWS Anda. Itu bukan batas tahunan dan tidak dapat ditingkatkan.
  - Anda dapat menjual hingga 50.000 USD dalam Instans Terpesan.
  - Anda dapat menjual hingga 5.000 USD Instans Terpesan.
- Tidak dapat mengubah — Anda tidak dapat mengubah iklan Anda secara langsung. Namun, Anda dapat mengubah daftar Anda dengan membatalkannya terlebih dahulu, lalu membuat daftar lain dengan parameter baru.
- Dapat membatalkan – Anda dapat membatalkan iklan Anda kapan saja, selama ada dalam status `active`. Anda tidak dapat membatalkan ikan jika sudah cocok atau sedang diproses untuk dijual. Jika beberapa instans dalam iklan Anda cocok dan Anda membatalkan iklan, hanya instans yang tidak cocok yang tersisa yang dihapus dari iklan.

### Mengiklankan Instans Terpesan Anda

Sebagai penjual terdaftar, Anda dapat memilih untuk menjual satu atau lebih dari Instans Terpesan Anda. Anda dapat memilih untuk menjual semuanya dalam satu iklan atau sebagian. Selain itu, Anda dapat mencantumkan Instans Terpesan dengan konfigurasi tipe instans, platform, dan cakupan apa pun.

Konsol menentukan harga yang disarankan. Konsol memeriksa penawaran yang cocok dengan Instans Terpesan Anda dan cocok dengan intans yang memiliki harga terendah. Jika tidak, konsol

menghitung harga yang disarankan berdasarkan biaya Instans Terpesan untuk sisa waktunya. Jika nilai yang dihitung kurang dari 1,01 USD, harga yang disarankan adalah 1,01 USD.

Jika Anda membatalkan iklan Anda dan sebagian dari iklan itu telah terjual, pembatalan tidak berlaku untuk porsi yang telah terjual. Hanya bagian yang tidak terjual dari listingan yang tidak lagi tersedia di Marketplace Instans Cadangan.

Console

Untuk mencantumkan Instans Cadangan di Marketplace Instans Cadangan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans Terpesan.
3. Pilih Instans Terpesan yang akan diiklankan, lalu pilih Tindakan, Jual Instans Terpesan.
4. Pada halaman Konfigurasikan Daftar Instans Terpesan Anda, tetapkan jumlah instans yang akan dijual dan harga dimuka untuk jangka waktu yang tersisa di kolom yang relevan. Lihat bagaimana nilai reservasi Anda berubah selama sisa jangka waktu dengan memilih panah di sebelah kolom Sisa Bulan.
5. Jika Anda adalah pengguna mahir dan ingin menyesuaikan harga, Anda dapat memasukkan nilai yang berbeda untuk bulan berikutnya. Untuk kembali ke penurunan harga linier default, pilih Atur ulang.
6. Pilih Lanjutkan setelah Anda selesai mengonfigurasi iklan Anda.
7. Konfirmasikan detail iklan Anda pada halaman Konfirmasi Iklan Instans Terpesan Anda, dan jika Anda puas, pilih Iklankan Instans Terpesan.

Untuk melihat daftar Anda di konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans Terpesan.
3. Pilih Instans Terpesan yang Anda iklankan dan pilih tab Iklan Saya di dekat bagian bawah halaman.

AWS CLI

Untuk mengelola Instans Cadangan di Pasar Instans Cadangan

1. Dapatkan daftar Instans Terpesan Anda dengan menggunakan perintah describe-reserved-instances. Perhatikan ID Instans Cadangan yang ingin Anda daftarkan.
2. Gunakan perintah create-reser ved-instances-listing. Anda harus menentukan ID Instans Terpesan, jumlah instans, dan jadwal harga.
3. Untuk melihat iklan Anda, gunakan perintah describe-reserved-instances-listings.
4. Untuk membatalkan iklan Anda, gunakan perintah cancel-reser ved-instances-listing.

PowerShell

Untuk mengelola Instans Cadangan di Pasar Instans Cadangan

1. Dapatkan daftar Instans Cadangan Anda dengan menggunakan Get-EC2ReservedInstance cmdlet. Perhatikan ID Instans Cadangan yang ingin Anda daftarkan.
2. Gunakan New-EC2ReservedInstancesListing cmdlet. Anda harus menentukan ID Instans Terpesan, jumlah instans, dan jadwal harga.
3. Untuk melihat iklan Anda, gunakan Get-EC2ReservedInstancesListing cmdlet.
4. Untuk membatalkan iklan Anda, gunakan Stop-EC2ReservedInstancesListing cmdlet.

Status iklan Instans Terpesan

Status Iklan pada tab Iklan saya dari halaman Instans Terpesan menampilkan status iklan Anda:

Informasi yang ditampilkan oleh Status Iklan adalah tentang status iklan Anda di Pasar Instans Terpesan. Hal ini berbeda dari informasi status yang ditampilkan oleh kolom Status di halaman Instans Terpesan. Informasi Status ini adalah tentang reservasi Anda.

- aktif—Iklan ini tersedia untuk dibeli.
- dibatalkan—Iklan dibatalkan dan tidak tersedia untuk dibeli di Pasar Instans Terpesan.
- ditutup — Instans Terpesan tidak diiklankan. Instans Terpesan mungkin saja`closed` karena penjualan iklan telah selesai.

Siklus hidup iklan

Jika semua instans dalam iklan Anda cocok dan terjual, tab Iklan Saya menunjukkan bahwa Jumlah instans total sama dengan jumlah yang tercantum dalam Terjual. Selain itu, tidak ada instans yang tersedia yang tersisa untuk iklan Anda, dan Status-nya adalah `closed`.

Jika hanya sebagian dari iklan Anda yang terjual, maka AWS instans Cadangan dibatalkan dalam daftar dan membuat jumlah Instans Cadangan yang sama dengan Instans Cadangan yang tersisa dalam hitungan. Jadi, ID iklan dan iklan yang diwakilinya, yang sekarang memiliki lebih sedikit reservasi untuk dijual, masih aktif.

Semua penjualan Instans Terpesan di masa mendatang dalam iklan ini diproses dengan cara ini. Ketika semua Instans Cadangan dalam daftar terjual, AWS tandai daftar sebagai`closed`.

Misalnya, Anda membuat iklan ID iklan Instans Terpesan 5ec28771-05ff-4b9b-aa31-9e57dexample dengan jumlah iklan 5.

Tab Iklan Saya di halaman konsol Instans Terpesan menampilkan iklan dengan cara ini:

ID daftar penawaran Instans Terpesan 5ec28771-05ff-4b9b-aa31-9e57dexample

- Jumlah total reservasi = 5
- Terjual = 0
- Tersedia = 5
- Status = aktif

Seorang pembeli membeli dua reservasi, sehingga tiga reservasi masih tersedia untuk dijual. Karena penjualan sebagian ini, AWS buat reservasi baru dengan hitungan tiga untuk mewakili sisa reservasi yang masih dijual.

Berikut tampilan iklan Anda di tab Iklan Saya:

ID daftar penawaran Instans Terpesan 5ec28771-05ff-4b9b-aa31-9e57dexample

- Jumlah total reservasi = 5
- Terjual = 2
- Tersedia = 3
- Status = aktif

Jika Anda membatalkan iklan Anda dan sebagian dari iklan itu telah terjual, pembatalan tidak berlaku untuk porsi yang telah terjual. Hanya bagian yang tidak terjual dari daftar penawaran yang tidak lagi tersedia di Marketplace Instans Terpesan.

Setelah Instans Terpesan Anda terjual

Saat Instans Cadangan Anda terjual, AWS mengirimkan pemberitahuan email kepada Anda. Setiap hari saat ada aktivitas apa pun, Anda menerima satu notifikasi email yang merekam semua aktivitas hari itu. Aktivitas dapat mencakup saat Anda membuat atau menjual iklan, atau saat AWS mengirim dana ke akun Anda.

Console

Untuk melacak status daftar Instans Cadangan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di halaman navigasi, pilih Instans Terpesan.
3. Pada tab Daftar Saya, temukan nilai Status Listing. Tab ini juga berisi informasi tentang jangka waktu, harga daftar, dan rincian berapa banyak instans dalam daftar yang tersedia, tertunda, dijual, dan dibatalkan.

AWS CLI

Untuk melacak status daftar Instans Cadangan

Gunakan perintah describe-reserved-instances-lists dengan filter yang sesuai untuk mendapatkan informasi tentang listing Anda.

```
aws ec2 describe-reserved-instances-listings
```

PowerShell

Untuk melacak status daftar Instans Cadangan

Gunakan Get-EC2ReservedInstancesListing cmdlet.

```
Get-EC2ReservedInstancesListing
```

Mendapatkan pembayaran

Segera setelah AWS menerima dana dari pembeli, pesan dikirim ke email akun pemilik terdaftar untuk Instans Cadangan yang dijual.

AWS mengirimkan transfer bank Automated Clearing House (ACH) ke rekening bank yang Anda tentukan. Biasanya, transfer ini dilakukan antara satu hingga tiga hari setelah Instans Terpesan Anda terjual. Pencairan dilakukan satu kali per hari. Anda akan menerima email dengan laporan pencairan setelah dana dikeluarkan. Perlu diingat bahwa Anda tidak dapat menerima pencairan sampai AWS menerima verifikasi dari bank Anda. Verifikasi ini bisa memakan waktu hingga dua minggu.

Instans Terpesan yang Anda jual terus muncul jika Anda menjelaskan Instans Terpesan Anda.

Anda menerima pencairan tunai untuk Instans Cadangan Anda melalui transfer bank langsung ke rekening bank Anda. AWS membebankan biaya layanan sebesar 12 persen dari total harga di muka setiap Instans Cadangan yang Anda jual di Marketplace Instans Cadangan.

Informasi yang dibagikan dengan pembeli

Saat Anda menjual di Marketplace Instans Cadangan, AWS bagikan nama resmi perusahaan Anda pada pernyataan pembeli sesuai dengan peraturan AS. Selain itu, jika pembeli menghubungi Dukungan karena pembeli perlu menghubungi Anda untuk mendapatkan faktur atau untuk beberapa alasan terkait pajak lainnya, AWS mungkin harus memberikan alamat email Anda kepada pembeli sehingga pembeli dapat menghubungi Anda secara langsung.

Untuk alasan serupa, kode pos pembeli dan informasi negara diberikan kepada penjual dalam laporan pencairan. Sebagai penjual, Anda mungkin memerlukan informasi ini untuk menyertai pajak transaksi yang diperlukan, yang Anda serahkan ke pemerintah (seperti pajak penjualan dan pajak pertambahan nilai).

AWS tidak dapat menawarkan nasihat pajak, tetapi jika spesialis pajak Anda menentukan bahwa Anda memerlukan informasi tambahan tertentu, hubungi Dukungan.

## Memodifikasi Instans Terpesan

Saat kebutuhan berubah, Anda dapat mengubah Instans Terpesan Standar atau Konvertibel dan terus mendapatkan keuntungan dari manfaat penagihan. Anda dapat memodifikasi atribut, seperti Zona Ketersediaan, ukuran instans (dalam keluarga dan generasi instans yang sama), serta cakupan Instans Terpesan Anda.

> **Note**
>
> Anda juga dapat menukar Instans Terpesan Konvertibel dengan Instans Terpesan Konvertibel lain dengan konfigurasi yang berbeda. Untuk informasi selengkapnya, lihat Menukar Instans Terpesan Konvertibel.

Anda dapat memodifikasi semua atau sebagian Instans Terpesan Anda. Anda dapat memisahkan Instans Terpesan asli menjadi dua atau lebih Instans Terpesan baru. Misalnya, jika Anda memiliki reservasi untuk 10 instans di `us-east-1a` dan memutuskan untuk memindahkan 5 instans ke `us-east-1b`, permintaan modifikasi menghasilkan dua reservasi baru: satu untuk 5 instans di `us-east-1a` dan yang lainnya untuk 5 instans di `us-east-1b`.

Anda juga dapat menggabungkan dua atau lebih Instans Terpesan menjadi satu Instans Terpesan. Misalnya, jika Anda memiliki empat Instans Terpesan `t2.small` dari masing-masing satu instans, Anda dapat menggabungkannya untuk membuat satu Instans Terpesan `t2.large`. Untuk informasi selengkapnya, lihat Dukungan untuk memodifikasi ukuran instans.

Setelah modifikasi, keuntungan dari Instans Terpesan hanya diterapkan pada instans yang cocok dengan parameter baru. Misalnya, jika Anda mengubah Zona Ketersediaan suatu reservasi, reservasi kapasitas dan keuntungan harga secara otomatis diterapkan ke penggunaan instans di Zona Ketersediaan yang baru. Instans yang tidak lagi sesuai dengan parameter baru akan dikenakan biaya sesuai On-Demand tarif, kecuali akun Anda memiliki reservasi lain yang berlaku.

Jika permintaan modifikasi Anda berhasil:

- Modifikasi reservasi akan langsung berlaku dan keuntungan harga diterapkan ke instans baru yang dimulai pada jam permintaan modifikasi. Misalnya, jika Anda berhasil memodifikasi reservasi pada pukul 21.15, keuntungan harga ditransfer ke instans baru Anda pada pukul 21.00. Anda bisa mendapatkan tanggal efektif dari Instans Terpesan yang dimodifikasi menggunakan perintah describe-reserved-instances.
- Reservasi asli telah pensiun. Tanggal berakhir reservasi adalah tanggal mulai reservasi baru, dan tanggal akhir reservasi baru sama dengan tanggal akhir Instans Terpesan asli. Jika Anda memodifikasi reservasi tiga tahun yang memiliki sisa 16 bulan dalam jangka waktunya, hasil reservasi yang dimodifikasi adalah reservasi 16 bulan dengan tanggal akhir yang sama seperti yang asli.
- Reservasi yang dimodifikasi mencantumkan harga tetap 0 USD dan bukan harga tetap dari reservasi asli.

- Harga tetap dari reservasi yang dimodifikasi tidak memengaruhi penghitungan tingkat harga diskon yang diterapkan ke akun Anda, yang didasarkan pada harga tetap dari reservasi asli.

Jika permintaan modifikasi gagal, Instans Terpesan Anda mempertahankan konfigurasi aslinya, dan langsung tersedia untuk permintaan modifikasi lainnya.

Tidak ada biaya untuk modifikasi, dan Anda tidak menerima tagihan atau faktur baru.

Anda dapat memodifikasi reservasi sesering apa pun, tetapi Anda tidak dapat mengubah atau membatalkan permintaan modifikasi yang menunggu keputusan setelah Anda mengirimkannya. Setelah modifikasi berhasil diselesaikan, Anda dapat mengirimkan permintaan modifikasi lain untuk membatalkan perubahan yang dibuat, jika perlu.

Daftar Isi


- Persyaratan dan pembatasan untuk modifikasi
- Dukungan untuk memodifikasi ukuran instans
- Mengirimkan permintaan modifikasi
- Memecahkan masalah permintaan modifikasi


## Persyaratan dan pembatasan untuk modifikasi

Anda dapat memodifikasi atribut ini sebagai berikut.

| Atribut yang dapat dimodifikasi | Platform yang didukung | Batasan dan pertimbangan |
|---|---|---|
| Ubah Zona Ketersediaan dalam Wilayah yang sama | Linux dan Windows | - |
| Ubah cakupan dari Zona Ketersediaan ke Wilayah dan sebaliknya | Linux dan Windows | Instans Terpesan zonal tercakup dalam Zona Ketersediaan dan kapasitas terpesan di Zona Ketersediaan tersebut. Jika Anda mengubah cakupan dari Zona Ketersedi aan ke Wilayah (dengan kata lain, dari zonal ke regional), |

| Atribut yang dapat dimodifikasi | Platform yang didukung | Batasan dan pertimbangan |
| --- | --- | --- |
| | | Anda kehilangan keuntungan reservasi kapasitas.<br><br>Instans Terpesan regional tercakup dalam Wilayah. Diskon Instans Terpesan Anda dapat diterapkan ke instans yang berjalan di Zona Ketersediaan mana pun di Wilayah tersebut. Selain itu, diskon Instans Terpesan berlaku untuk penggunaa n instans di semua ukuran dalam keluarga instans yang dipilih. Jika Anda mengubah cakupan dari Wilayah ke Zona Ketersediaan (dengan kata lain, dari regional ke zonal), Anda kehilangan fleksibil itas Zona Ketersediaan dan fleksibilitas ukuran instans (jika berlaku).<br><br>Untuk informasi selengkap nya, lihat Bagaimana diskon Instans Cadangan diterapkan. |

| Atribut yang dapat dimodifikasi | Platform yang didukung | Batasan dan pertimbangan |
| --- | --- | --- |
| Ubah ukuran instans dalam keluarga dan generasi instans yang sama | Linux/UNIX hanya<br><br>Fleksibilitas ukuran instans tidak tersedia untuk Instans Terpesan di platform lain, yang mencakup Linux dengan SQL Server Standard, Linux dengan SQL Server Web, Linux dengan SQL Server Enterprise, Red Hat Enterprise Linux, SUSE Linux, Windows, Windows dengan SQL Standard, Windows dengan SQL Server Enterprise, dan Windows dengan SQL Server Web. | Reservasi harus menggunakan penghunian default. Beberapa keluarga instans tidak didukung karena tidak ada ukuran lain yang tersedia. Untuk informasi selengkapnya, lihat Dukungan untuk memodifikasi ukuran instans |

Persyaratan

Amazon EC2 memproses permintaan modifikasi jika ada kapasitas yang memadai untuk konfigurasi baru Anda (jika berlaku), dan jika kondisi berikut terpenuhi:

- Instans Terpesan tidak dapat dimodifikasi sebelum atau pada saat yang sama Anda membelinya
- Instans Terpesan harus aktif
- Tidak ada permintaan modifikasi yang menunggu keputusan
- Instans Terpesan tidak terdaftar di Marketplace Instans Terpesan
- Harus ada kecocokan antara jejak ukuran instans reservasi asli dan konfigurasi baru. Untuk informasi selengkapnya, lihat Dukungan untuk memodifikasi ukuran instans.
- Instans Terpesan asli adalah semua Instans Terpesan Standar atau semua Instans Terpesan Konvertibel, bukan beberapa tipe Instans Terpesan
- Instans Terpesan asli harus kedaluwarsa dalam jam yang sama, jika instans Terpesan tersebut adalah Instans Terpesan Standar

- Untuk memodifikasi ukuran instance, Instans Cadangan harus mendukung fleksibilitas ukuran instance. Untuk daftar Instans Cadangan yang tidak mendukung fleksibilitas ukuran instans, lihatFleksibilitas ukuran instans.

Dukungan untuk memodifikasi ukuran instans

Anda dapat memodifikasi ukuran instans dari Instans Terpesan jika persyaratan berikut terpenuhi.

Persyaratan

- Platformnya adalah Linux/UNIX.
- Anda harus memilih ukuran instans lain dalam keluarga instans yang sama (ditunjukkan dengan huruf, misalnya, T) dan generas yang sama (ditunjukkan oleh angka, misalnya, 2).

  Misalnya, Anda dapat memodifikasi Instans Terpesan dari `t2.small` ke `t2.large` karena keduanya berada dalam keluarga dan generasi T2 yang sama. Namun, Anda tidak dapat memodifikasi Instans Terpesan dari T2 ke M2 atau dari T2 ke T3, karena dalam kedua contoh ini, keluarga dan generasi instans target tidak sama dengan Instans Terpesan asli.
- Anda dapat mengubah ukuran instance dari Instans Cadangan hanya jika mendukung fleksibilitas ukuran instans. Untuk daftar Instans Cadangan yang tidak mendukung fleksibilitas ukuran instans, lihatFleksibilitas ukuran instans.
- Anda tidak dapat mengubah ukuran instance Instans Cadangan untuk `t1.micro` instans, karena hanya `t1.micro` memiliki satu ukuran.
- Instans Terpesan yang asli dan baru harus memiliki jejak ukuran instans yang sama.

Daftar Isi


- Jejak ukuran instans
- Faktor normalisasi untuk instans bare metal


Jejak ukuran instans

Setiap Instans Terpesan memiliki jejak ukuran instans, yang ditentukan oleh faktor normalisasi ukuran instans dan jumlah instans dalam reservasi. Saat Anda memodifikasi ukuran instans dalam Instans Terpesan, jejak konfigurasi baru harus cocok dengan konfigurasi asli, jika tidak, permintaan modifikasi tidak akan diproses.

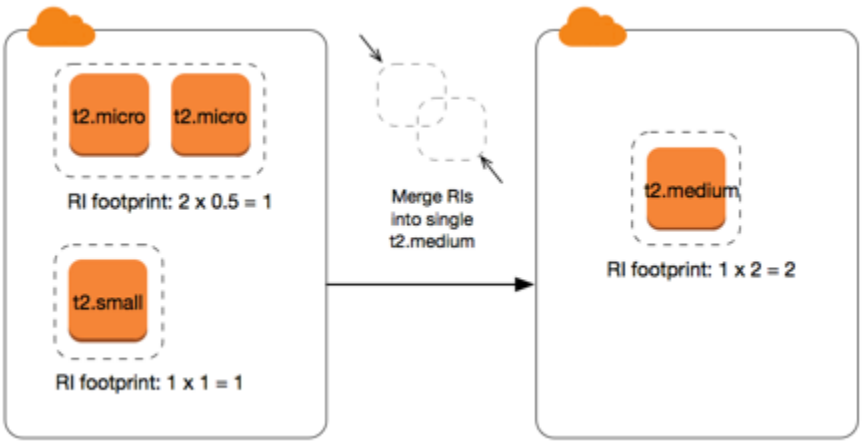
Untuk menghitung jejak ukuran instans dari Instans Terpesan, kalikan jumlah instans dengan faktor normalisasi. Di konsol Amazon EC2, faktor normalisasi diukur dalam satuan. Tabel berikut menjelaskan faktor normalisasi untuk ukuran instans dalam suatu keluarga instans. Misalnya, `t2.medium` memiliki faktor normalisasi 2, jadi reservasi untuk empat instans `t2.medium` memiliki jejak dari 8 unit.

| Ukuran instans | Faktor normalisasi |
|---|---|
| nano | 0.25 |
| micro | 0.5 |
| small | 1 |
| medium | 2 |
| large | 4 |
| xlarge | 8 |
| 2xlarge | 16 |
| 3xlarge | 24 |
| 4xlarge | 32 |
| 6xlarge | 48 |
| 8xlarge | 64 |
| 9xlarge | 72 |
| 10xlarge | 80 |
| 12xlarge | 96 |
| 16xlarge | 128 |
| 18xlarge | 144 |
| 24xlarge | 192 |

| Ukuran instans | Faktor normalisasi |
|---|---|
| 32xlarge | 256 |
| 48xlarge | 384 |
| 56xlarge | 448 |
| 96xlarge | 768 |
| 112xlarge | 896 |

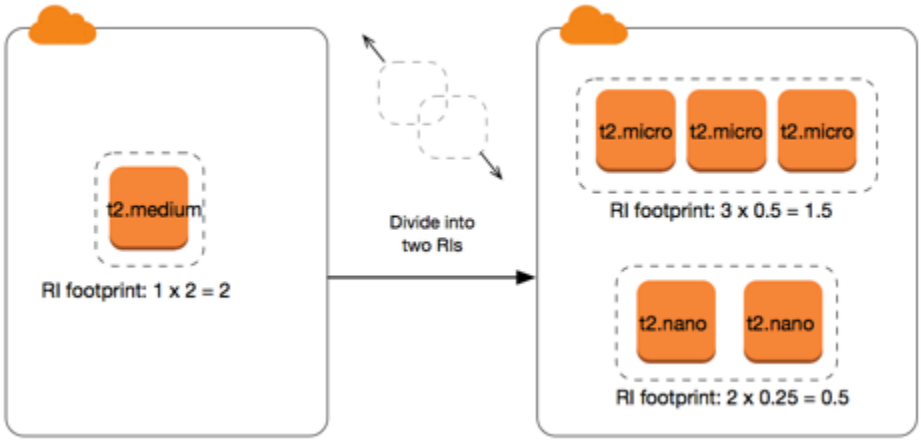
Anda dapat mengalokasikan reservasi Anda ke dalam ukuran instans yang berbeda di seluruh keluarga instans yang sama selama jejak ukuran instans reservasi Anda tetap sama. Misalnya, Anda dapat membagi reservasi untuk satu instans (1 @ 4 unit) `t2.large` menjadi empat instans (4 @ 1 unit) `t2.small`. Demikian pula, Anda dapat menggabungkan reservasi untuk empat instans `t2.small` menjadi satu instans `t2.large`. Namun, Anda tidak dapat mengubah reservasi untuk dua instans `t2.small` menjadi satu instans `t2.large` karena jejak reservasi baru (4 unit) lebih besar dari jejak reservasi asli (2 unit).

Dalam contoh berikut, Anda memiliki reservasi dengan dua instans `t2.micro` (1 unit) dan reservasi dengan satu instans `t2.small` (1 unit). Jika Anda menggabungkan kedua reservasi ini menjadi satu reservasi dengan satu instans `t2.medium` (2 unit), maka jejak dari reservasi baru sama dengan jejak dari reservasi gabungan.




Anda juga dapat memodifikasi reservasi untuk membaginya menjadi dua reservasi atau lebih. Dalam contoh berikut, Anda memiliki reservasi dengan satu instans `t2.medium` (2 unit). Anda dapat

membagi reservasi menjadi dua, satu dengan dua instans `t2.nano` (0,5 unit) dan yang lainnya dengan tiga instans `t2.micro` (1,5 unit).




Faktor normalisasi untuk instans bare metal

Anda dapat memodifikasi reservasi dengan instans `metal` menggunakan ukuran lain dalam keluarga instans yang sama. Demikian pula, Anda dapat memodifikasi reservasi dengan instans selain instans bare metal menggunakan ukuran `metal` dalam keluarga instans yang sama. Umumnya, instans bare metal memiliki ukuran yang sama dengan ukuran instans terbesar yang tersedia dalam keluarga instans yang sama. Misalnya, sebuah instans `i3.metal` berukuran sama dengan instans `i3.16xlarge`, sehingga keduanya memiliki faktor normalisasi yang sama.

Tabel berikut menjelaskan faktor normalisasi untuk ukuran instans bare metal dalam keluarga instans yang memiliki instans bare metal. Faktor normalisasi untuk instans `metal` bergantung pada keluarga instans, tidak seperti ukuran instans lainnya.

| Ukuran instans | Faktor normalisasi |
| --- | --- |
| `a1.metal` | 32 |
| `m5zn.metal` \| `x2iezn.metal` `z1d.metal` | 96 |
| `c6g.metal` \| `c6gd.metal` \| `i3.metal` \| `m6g.metal` \| `m6gd.metal` \| `r6g.metal` \| `r6gd.metal` \| `x2gd.metal` | 128 |
| `c5n.metal` | 144 |

| Ukuran instans | Faktor normalisasi |
| --- | --- |
| c5.metal \| c5d.metal \| i3en.metal \| m5.metal \| m5d.metal \| m5dn.metal \| m5n.metal \| r5.metal \| r5b.metal \| r5d.metal \| r5dn.metal \| r5n.metal | 192 |
| c6i.metal \| c6id.metal \| m6i.metal \| m6id.metal \| r6d.metal \| r6id.metal | 256 |
| u-18tb1.metal \| u-24tb1.metal | 448 |
| u-6tb1.metal \| u-9tb1.metal \| u-12tb1.metal | 896 |

Misalnya, instans `i3.metal` memiliki faktor normalisasi 128. Jika Anda membeli Instans `i3.metal` Cadangan Amazon sewa Linux/Unix default, Anda dapat membagi reservasi sebagai berikut:

- An `i3.16xlarge` adalah ukuran yang sama dengan `i3.metal` instance, jadi faktor normalisasinya adalah 128 (128/1). Reservasi untuk satu instans `i3.metal` dapat dimodifikasi menjadi satu instans `i3.16xlarge`.
- An `i3.8xlarge` adalah setengah ukuran `i3.metal` instance, jadi faktor normalisasinya adalah 64 (128/2). Reservasi untuk satu instans `i3.metal` dapat dibagi menjadi dua instans `i3.8xlarge`.
- An `i3.4xlarge` adalah seperempat ukuran `i3.metal` instance, jadi faktor normalisasinya adalah 32 (128/4). Reservasi untuk satu instans `i3.metal` dapat dibagi menjadi empat instans `i3.4xlarge`.

Mengirimkan permintaan modifikasi

Sebelum mengubah Instans Cadangan, pastikan Anda telah membaca batasan yang berlaku. Sebelum Anda mengubah ukuran instans, hitung total ukuran instance dari reservasi asli yang ingin Anda ubah dan pastikan ukuran tersebut cocok dengan total ukuran instance dari konfigurasi baru Anda.

Console

Untuk mengubah Instans Cadangan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di halaman Instans Terpesan, pilih satu atau beberapa Instans Terpesan yang akan dimodifikasi, dan pilih Tindakan, Modifikasi Instans Terpesan.

   Jika Instans Terpesan Anda tidak dalam keadaan aktif atau tidak dapat dimodifikasi, Modifikasi Instans Terpesan akan dinonaktifkan.
3. Entri pertama dalam tabel modifikasi menampilkan atribut dari Instans Terpesan yang dipilih, dan setidaknya satu konfigurasi target di bawahnya. Kolom Unit menampilkan jejak ukuran instans secara total. Pilih Tambahkan untuk setiap konfigurasi baru yang akan ditambahkan. Modifikasi atribut seperlunya untuk setiap konfigurasi.

   - Cakupan: Pilih apakah konfigurasi berlaku untuk sebuah Zona Ketersediaan atau seluruh Wilayah.
   - Zona Ketersediaan: Pilih Zona Ketersediaan yang dibutuhkan. Tidak berlaku untuk Instans Terpesan wilayah.
   - Jenis instans: Pilih jenis instans yang diperlukan. Konfigurasi gabungan harus mempunyai jejak ukuran instans yang sama dengan konfigurasi asli Anda.
   - Jumlah: Tentukan jumlah instans. Untuk membagi Instans Terpesan ke dalam banyak konfigurasi, kurangi jumlah, pilih Tambahkan, dan tentukan jumlah untuk konfigurasi tambahan. Misalnya, jika Anda memiliki konfigurasi tunggal dengan jumlah 10, Anda dapat mengubah jumlahnya menjadi 6 dan menambahkan konfigurasi dengan jumlah 4. Proses ini memensiunkan Instans Terpesan asli setelah Instans Terpesan baru diaktifkan.
4. Pilih Lanjutkan.
5. Untuk mengonfirmasi pilihan modifikasi setelah Anda selesai menentukan konfigurasi target Anda, pilih Kirim modifikasi.
6. Anda dapat menentukan status permintaan modifikasi dengan melihat kolom Status di layar Instans Terpesan. Berikut ini adalah beberapa kemungkinan status.

   - aktif (modifikasi tertunda) — Status transisi untuk Instans Terpesan asli
   - pensiun (modifikasi tertunda) — Status transisi untuk Instans Terpesan asli sementara Instans Terpesan baru sedang dibuat
   - pensiun — Instans Terpesan berhasil dimodifikasi dan diganti

- aktif — Salah satu dari berikut ini:
  - Instans Terpesan baru dibuat dari permintaan modifikasi yang berhasil
  - Instans Terpesan Asli setelah permintaan modifikasi gagal

AWS CLI

Untuk mengubah Instans Cadangan

Gunakan perintah modify- reserved-instance. Anda dapat memberikan detail konfigurasi dalam file JSON.

```
aws ec2 modify-reserved-instances \
    --reserved-instances-ids b847fa93-e282-4f55-b59a-1342f5bd7c02 \
    --target-configurations file://configuration.json
```

Untuk mendapatkan status permintaan modifikasi Anda

Gunakan perintah describe-reser ved-instances-. Statusnya adalah`processing`,`fulfilled`, atau`failed`.

```
aws ec2 describe-reserved-instances-modifications \
    --reserved-instances-modification-ids rimod-d3ed4335-b1d3-4de6-ab31-0f13aaf46687
 \
    --query ReservedInstancesModifications[].Status
```

PowerShell

Untuk mengubah Instans Cadangan

Gunakan Edit-EC2ReservedInstance cmdlet. Anda dapat memberikan detail konfigurasi dalam objek tipe`Amazon.EC2.Model.ReservedInstancesConfiguration`.

```
Edit-EC2ReservedInstance `
    -ReservedInstancesId b847fa93-e282-4f55-b59a-1342f5bd7c02 `
    -TargetConfiguration $configuration
```

Untuk mendapatkan status permintaan modifikasi Anda

Gunakan Get-EC2ReservedInstancesModification cmdlet. Statusnya adalah`processing`,`fulfilled`, atau`failed`.

```
Get-EC2ReservedInstancesModification `
    -ReservedInstancesModificationId rimod-d3ed4335-b1d3-4de6-ab31-0f13aaf46687 | `
    Select Status
```

Memecahkan masalah permintaan modifikasi

Jika pengaturan konfigurasi target yang Anda minta unik, Anda menerima pesan bahwa permintaan Anda sedang diproses. Pada titik ini, Amazon EC2 hanya menentukan bahwa parameter permintaan modifikasi Anda valid. Permintaan modifikasi Anda masih bisa gagal selama pemrosesan karena ketidaktersediaan kapasitas.

Dalam beberapa situasi, Anda mungkin mendapatkan pesan yang menunjukkan permintaan modifikasi yang tidak selesai atau gagal alih-alih konfirmasi. Gunakan informasi dalam pesan tersebut sebagai titik awal untuk mengirim ulang permintaan modifikasi lainnya. Pastikan Anda telah membaca pembatasan berlaku sebelum mengirimkan permintaan.

Tidak semua Instans Terpesan yang dipilih dapat diproses untuk modifikasi

Amazon EC2 mengidentifikasi dan mencantumkan daftar Instans Terpesan yang tidak dapat dimodifikasi. Jika Anda menerima pesan seperti ini, buka halaman Instans Terpesan di konsol Amazon EC2 dan periksa informasi untuk Instans Terpesan.

Kesalahan dalam memproses permintaan modifikasi Anda

Anda mengirimkan satu atau lebih Instans Terpesan untuk modifikasi dan tidak ada permintaan Anda yang dapat diproses. Tergantung jumlah reservasi yang dimodifikasi, Anda bisa mendapatkan versi berbeda dari pesan tersebut.

Amazon EC2 menampilkan alasan mengapa permintaan Anda tidak dapat diproses. Misalnya, Anda mungkin telah menetapkan konfigurasi target yang sama—kombinasi dari Zona Ketersediaan dan platform—untuk satu atau beberapa subset Instans Terpesan yang Anda modifikasi. Coba kirimkan permintaan modifikasi lagi, tetapi pastikan bahwa detail instans dari reservasi cocok, dan bahwa konfigurasi target untuk semua subset yang dimodifikasi adalah unik.

## Menukar Instans Terpesan Konvertibel

Anda dapat menukar satu atau beberapa Instans Terpesan Konvertibel dengan Instans Terpesan Konvertibel lainnya dengan konfigurasi yang berbeda, termasuk keluarga instans, sistem operasi, dan penghunian. Tidak ada batasan terkait frekuensi penukaran, selama Instans Terpesan

Konvertibel baru memiliki nilai yang sama atau lebih tinggi dari Instans Terpesan Konvertibel asli yang Anda tukar.

Saat Anda menukar Instans Terpesan Konvertibel, jumlah instans untuk reservasi Anda saat ini ditukar dengan sejumlah instans yang mencakup nilai yang sama atau lebih tinggi dari konfigurasi Instans Terpesan Konvertibel baru. Amazon EC2 menghitung jumlah Instans Terpesan yang dapat Anda terima sebagai hasil dari pertukaran.

Anda tidak dapat menukar Instans Terpesan Standar, tetapi Anda dapat memodifikasinya. Untuk informasi selengkapnya, lihat Memodifikasi Instans Terpesan.

Daftar Isi


- Persyaratan untuk menukar Instans Terpesan Konvertibel
- Menghitung pertukaran Instans Terpesan Konvertibel
- Menggabungkan Instans Terpesan Konvertibel
- Menukar sebagian dari Instans Terpesan Konvertibel
- Mengirimkan permintaan pertukaran


## Persyaratan untuk menukar Instans Terpesan Konvertibel

Jika kondisi berikut terpenuhi, Amazon EC2 memproses permintaan pertukaran Anda. Instans Terpesan Konvertibel Anda harus:

- Aktif
- Tidak menunggu permintaan pertukaran sebelumnya
- Memiliki setidaknya 24 jam yang tersisa sebelum kedaluwarsa

Aturan-aturan berikut berlaku:

- Instans Cadangan Konvertibel harus ditukar dengan Instans Cadangan Konvertibel lainnya yang saat ini ditawarkan oleh AWS.
- Instans Terpesan Konvertibel dikaitkan dengan Wilayah tertentu, yang ditetapkan selama jangka waktu reservasi. Anda tidak dapat menukar Instans Cadangan Konvertibel dengan Instans Cadangan Konvertibel di Wilayah yang berbeda.
- Untuk menukar Instans Cadangan Konvertibel zona, AWS harus memiliki kapasitas yang cukup untuk jenis instans baru di Wilayah.

- Anda dapat menukar satu atau beberapa Instans Terpesan Konvertibel untuk satu Instans Terpesan Konvertibel yang baru saja dalam satu waktu.
- Untuk menukar sebagian Instans Terpesan Konvertibel, Anda dapat memodifikasinya menjadi dua atau lebih reservasi, lalu menukar satu atau lebih reservasi dengan Instans Terpesan Konvertibel yang baru. Untuk informasi selengkapnya, lihat Menukar sebagian dari Instans Terpesan Konvertibel. Untuk informasi selengkapnya tentang memodifikasi Instans Terpesan, lihat Memodifikasi Instans Terpesan.
- Semua Instans Terpesan Konvertibel di Muka dapat ditukar dengan Instans Terpesan Konvertibel Sebagian di Muka, dan sebaliknya.

> **Note**
>
> Jika total pembayaran dimuka yang diperlukan untuk pertukaran (biaya true-up) kurang dari $0,00, AWS secara otomatis memberi Anda jumlah instans di Instans Cadangan Konvertibel yang memastikan bahwa biaya true-up adalah $0,00 atau lebih.

> **Note**
>
> Jika nilai total (harga dimuka + harga per jam* jumlah sisa jam) dari Instans Cadangan Konversi baru kurang dari nilai total Instans Cadangan Konversi yang ditukar, AWS secara otomatis memberi Anda jumlah instans di Instans Cadangan Konversi yang dikonversi yang memastikan bahwa nilai totalnya sama atau lebih tinggi dari Instans Cadangan Konversi yang ditukar.

- Untuk mendapatkan keuntungan dari harga yang lebih baik, Anda dapat menukar Instans Terpesan Konvertibel Tanpa Biaya di Muka dengan Instans Terpesan Konvertibel Biaya Semua di muka atau Sebagian di Muka.
- Anda tidak dapat menukar Semua Instans Cadangan Konversi di Muka dan Sebagian yang Dapat Dikonversi di Muka dengan Instans Cadangan Tanpa Konversi di Muka.
- Anda dapat menukar Instans Terpesan Konvertibel Tanpa Biaya di Muka dengan Instans Terpesan Konvertibel Tanpa Biaya di Muka lainnya hanya jika harga per jam Instans Terpesan Konvertibel yang baru sama atau lebih tinggi dari harga per jam Instans Terpesan Konvertibel yang ditukar.

> **Note**
>
> Jika nilai total (harga per jam * jumlah jam yang tersisa) dari Instans Terpesan Konvertibel yang baru kurang dari nilai total Instans Terpesan Konvertibel yang ditukar, AWS secara otomatis memberi Anda sejumlah instans dalam Instans Terpesan Konvertibel yang memastikan bahwa total instans nilainya sama atau lebih tinggi dari Instans Terpesan Konvertibel yang ditukar.

- Jika Anda menukar banyak Instans Terpesan Konvertibel yang memiliki tanggal kedaluwarsa berbeda, tanggal kedaluwarsa untuk Instans Terpesan Konvertibel yang baru adalah tanggal terjauh di masa mendatang.
- Jika Anda menukar satu Instans Terpesan Konvertibel, instans tersebut harus memiliki jangka waktu yang sama (1 tahun atau 3 tahun) dengan Instans Terpesan Konvertibel yang baru. Jika Anda menggabungkan beberapa Instans Terpesan Konvertibel dengan jangka waktu berbeda, Instans Terpesan Konvertibel yang baru memiliki jangka waktu 3 tahun. Untuk informasi selengkapnya, lihat Menggabungkan Instans Terpesan Konvertibel.
- Saat Amazon EC2 menukar Instans Terpesan Konvertibel, mereka memensiunkan reservasi terkait, dan mentransfer tanggal akhir ke reservasi baru. Setelah pertukaran, Amazon EC2 menetapkan tanggal akhir untuk reservasi lama dan tanggal mulai untuk reservasi baru sama dengan tanggal pertukaran. Misalnya, jika Anda menukar reservasi tiga tahun yang memiliki sisa jangka waktu 16 bulan, reservasi baru adalah reservasi 16 bulan dengan tanggal akhir yang sama dengan reservasi dari Instans Terpesan Konvertibel yang Anda tukarkan.

### Menghitung pertukaran Instans Terpesan Konvertibel

Bertukar Instans Terpesan Konvertibel bersifat gratis. Namun, Anda mungkin diharuskan untuk membayar biaya true-up, yang merupakan biaya di muka yang dihitung prorata dari selisih antara Instans Terpesan Konvertibel yang Anda miliki dan Instans Terpesan Konvertibel baru yang Anda terima dari pertukaran tersebut.

Setiap Instans Terpesan Konvertibel memiliki nilai daftar. Nilai daftar ini dibandingkan dengan nilai daftar Instans Cadangan Konversi yang ingin Anda tentukan berapa banyak reservasi instans yang dapat Anda terima dari bursa.

Sebagai contoh: Anda memiliki Instans Terpesan Konvertibel dengan nilai daftar 1 x 35 USD yang ingin Anda tukarkan dengan tipe instans baru dengan nilai daftar 10 USD.

```
$35/$10 = 3.5
```

Anda dapat menukar Instans Terpesan Konvertibel dengan tiga Instans Terpesan Konvertibel senilai 10 USD. Membeli setengah reservasi tidak dimungkinkan; oleh karena itu Anda harus membeli Instans Terpesan Konvertibel tambahan untuk menutupi sisanya:

```
3.5 = 3 whole Convertible Reserved Instances + 1 additional Convertible Reserved
 Instance
```

Instans Terpesan Konvertibel keempat memiliki tanggal berakhir yang sama dengan tiga lainnya. Jika Anda menukar Instans Terpesan Konvertibel secara Sebagian atau secara Penuh di Muka, Anda membayar biaya true-up untuk reservasi keempat. Jika biaya di muka yang tersisa dari Instans Terpesan Konvertibel adalah 500 USD, dan reservasi baru biasanya akan dikenakan biaya 600 USD secara prorata, maka Anda akan dikenai biaya 100 USD.

```
$600 prorated upfront cost of new reservations - $500 remaining upfront cost of old
 reservations = $100 difference
```

Menggabungkan Instans Terpesan Konvertibel

Jika Anda menggabungkan dua atau lebih Instans Terpesan Konvertibel, jangka waktu Instans Terpesan Konvertibel yang baru harus sama dengan Instans Terpesan Konvertibel yang lama, atau yang tertinggi dari Instans Terpesan Konvertibel. Tanggal kedaluwarsa untuk Instans Terpesan Konvertibel yang baru adalah tanggal kedaluwarsa yang terjauh di masa mendatang.

Misalnya, Anda memiliki Instans Terpesan Konvertibel berikut ini di akun Anda:

| ID Instans Terpesan | Jangka waktu | Tanggal kedaluwarsa |
|---|---|---|
| aaaa1111 | 1 tahun | 31/12/2018 |
| bbbb2222 | 1 tahun | 31/07/2018 |
| cccc3333 | 3 tahun | 30/06/2018 |
| dddd4444 | 3 tahun | 31/12/2019 |

- Anda dapat menggabungkan `aaaa1111` dan `bbbb2222`, serta menukarnya dengan Instans Terpesan Konvertibel 1 tahun. Anda tidak dapat menukarnya dengan Instans Terpesan Konvertibel 3 tahun. Tanggal kedaluwarsa Instans Terpesan Konvertibel yang baru adalah 31-12-2018.
- Anda dapat menggabungkan `bbbb2222` dan `cccc3333` serta menukarnya dengan Instans Terpesan Konvertibel 3 tahun. Anda tidak dapat menukarnya dengan Instans Terpesan Konvertibel 1 tahun. Tanggal kedaluwarsa Instans Terpesan Konvertibel yang baru adalah 31-07-2018.
- Anda dapat menggabungkan `cccc3333` dan `dddd4444` serta menukarnya dengan Instans Terpesan Konvertibel 3 tahun. Anda tidak dapat menukarnya dengan Instans Terpesan Konvertibel 1 tahun. Tanggal kedaluwarsa Instans Terpesan Konvertibel yang baru adalah 31-12-2019.

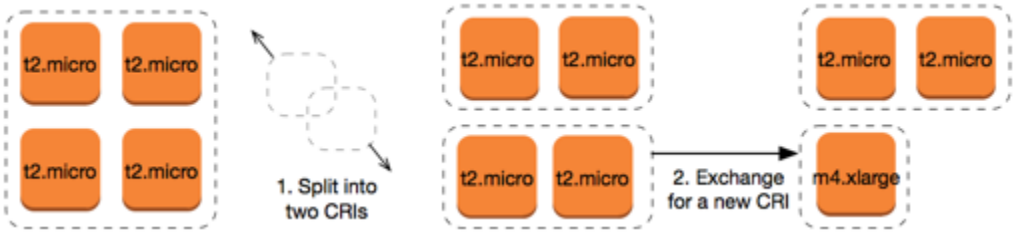
### Menukar sebagian dari Instans Terpesan Konvertibel

Anda dapat menggunakan proses modifikasi untuk membagi Instans Terpesan Konvertibel menjadi reservasi yang lebih kecil, kemudian menukar satu atau lebih reservasi baru untuk Instans Terpesan Konvertibel baru. Contoh berikut menunjukkan cara untuk melakukannya.

#### Example Contoh: Instans Terpesan Konvertibel dengan lebih dari satu instans

Dalam contoh ini, Anda memiliki file Instans Terpesan Konvertibel `t2.micro` dengan empat instans dalam reservasi. Untuk menukar dua instans `t2.micro` dengan instans `m4.xlarge`:

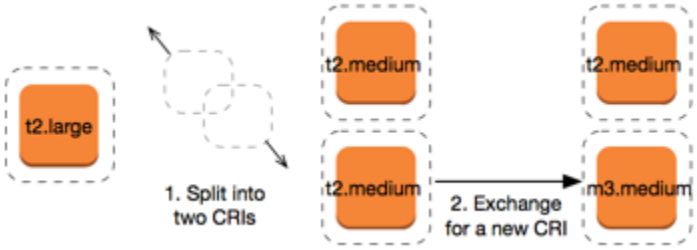
1. Modifikasi Instans Terpesan Konvertibel `t2.micro` dengan membaginya menjadi dua Instans Terpesan Konvertibel `t2.micro`, masing-masing dua instans.
2. Tukar salah satu Instans Terpesan Konvertibel `t2.micro` yang baru dengan Instans Terpesan Konvertibel `m4.xlarge`.




#### Example Contoh: Instans Terpesan Konvertibel dengan satu instans

Dalam contoh ini, Anda memiliki Instans Terpesan Konvertibel `t2.large`. Untuk mengubahnya menjadi instans `t2.medium` dan instans `m3.medium` yang lebih kecil:

1. Modifikasi Instans Terpesan Konvertibel `t2.large` dengan membaginya menjadi dua Instans Terpesan Konvertibel `t2.medium`. Satu instans `t2.large` memiliki jejak ukuran instans yang sama dengan dua instans `t2.medium`.
2. Tukar salah satu Instans Terpesan Konvertibel `t2.medium` yang baru dengan Instans Terpesan Konvertibel `m3.medium`.




Untuk informasi selengkapnya, lihat Dukungan untuk memodifikasi ukuran instans dan Mengirimkan permintaan pertukaran.

Mengirimkan permintaan pertukaran

Anda dapat menukar Instans Cadangan Konvertibel Anda. Instans Cadangan yang ditukar akan dihentikan.

Console

Untuk menukar Instans Cadangan yang Dapat Dikonversi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Instans Terpesan, pilih Instans Terpesan Konvertibel yang akan ditukar, dan pilih Tindakan, Tukar Instans Terpesan.
3. Pilih atribut dari konfigurasi yang diinginkan, dan pilih Temukan Penawaran.
4. Pilih Instans Terpesan Konvertibel baru. Di bagian bawah layar, Anda dapat melihat jumlah Instans Terpesan yang Anda terima untuk pertukaran tersebut, dan biaya tambahannya.
5. Jika Anda telah memilih Instans Terpesan Konvertibel yang memenuhi kebutuhan Anda, pilih Tinjau.
6. Pilih Tukar, kemudian Tutup.

AWS CLI

Untuk menukar Instans Cadangan yang Dapat Dikonversi

1. Temukan Instans Cadangan Konversi baru yang memenuhi kebutuhan Anda dengan menggunakan perintah describe-reser ved-instances-offers.
2. Dapatkan penawaran untuk pertukaran dengan menggunakan perintah get-reserved-instances-exchange-quote. Ini termasuk jumlah Instans Cadangan yang Anda dapatkan dari bursa, dan biaya aktual untuk pertukaran:
3. Lakukan pertukaran dengan menggunakan perintah accept-reser ved-instances-exchange-quote.

PowerShell

Untuk menukar Instans Cadangan yang Dapat Dikonversi

1. Temukan Instans Cadangan Konversi baru yang memenuhi kebutuhan Anda dengan menggunakan Get-EC2ReservedInstancesOffering cmdlet.
2. Dapatkan penawaran untuk pertukaran dengan menggunakan GetEC2-ReservedInstancesExchangeQuote cmdlet. Ini termasuk jumlah Instans Cadangan yang Anda dapatkan dari bursa, dan biaya aktual untuk pertukaran:
3. Lakukan pertukaran dengan menggunakan Approve-EC2ReservedInstancesExchangeQuote cmdlet

## Kuota Instans Terpesan

Anda dapat membeli Instans Terpesan baru setiap bulan. Jumlah Instans Terpesan baru yang dapat Anda beli setiap bulan ditentukan oleh kuota bulanan Anda, sebagai berikut:

| Deskripsi kuota | Kuota default |
| --- | --- |
| Instans Terpesan regional baru | 20 per Wilayah per bulan |
| Instans Terpesan zonal baru | 20 per Zona Ketersediaan per bulan |

Misalnya, di Wilayah dengan tiga Zona Ketersediaan, kuota default-nya adalah 80 Instans Terpesan per bulan, dihitung sebagai berikut:

- 20 Instans Terpesan regional untuk Wilayah
- Ditambah 60 Instans Terpesan zonal (20 untuk masing-masing dari tiga Zona Ketersediaan)

Instans di `running` negara bagian diperhitungkan dalam kuota Anda. Instans yang berada di negara `pending` bagian`stopping`,`stopped`,, dan `hibernated` tidak dihitung dalam kuota Anda.

Melihat jumlah Instans Terpesan yang telah Anda beli

Jumlah Instans Terpesan yang Anda beli ditunjukkan oleh bidang Jumlah instans (konsol) atau parameter `InstanceCount` (AWS CLI). Saat Anda membeli Instans Terpesan baru, kuota diukur terhadap jumlah instans total. Misalnya, jika Anda membeli satu konfigurasi Instans Terpesan dengan satu instans berjumlah 10, pembelian diperhitungkan dengan kuota Anda sebagai 10, bukan 1.

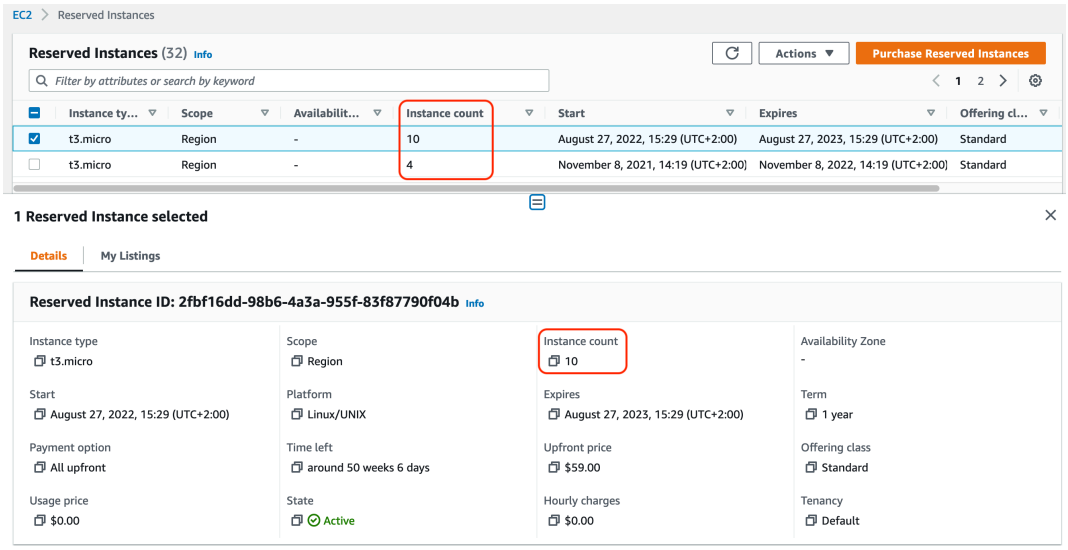
Anda dapat melihat jumlah Instans Terpesan yang telah dibeli menggunakan Amazon EC2 atau AWS CLI.

Console

Untuk melihat jumlah Instans Terpesan yang telah Anda beli

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans Terpesan.
3. Pilih konfigurasi Instans Terpesan dari tabel, dan periksa bidang Jumlah instans.

   Pada tangkapan layar berikut, baris yang dipilih mewakili konfigurasi satu Instans Terpesan untuk tipe instans `t3.micro`. Kolom Jumlah instans dalam tampilan tabel dan bidang Jumlah instans dalam tampilan detail (diuraikan dalam tangkapan layar) menunjukkan bahwa ada 10 Instans Terpesan untuk konfigurasi ini.




## AWS CLI

### Untuk melihat jumlah Instans Terpesan yang telah Anda beli

Gunakan perintah describe-reserved-instance dan tentukan ID konfigurasi Instance Cadangan.

```
aws ec2 describe-reserved-instances \
    --reserved-instances-ids a1b2c3d4-5678-90ab-cdef-EXAMPLE11111 \
    --output table
```

Berikut ini adalah output contoh. Bid `InstanceCount` ang menunjukkan bahwa ada 10 Instans Cadangan untuk konfigurasi ini.

```
-------------------------------------------------------------------
|                   DescribeReservedInstances                     |
+-----------------------------------------------------------------+
||                       ReservedInstances                       ||
|+----------------------+----------------------------------------+|
||  CurrencyCode        |  USD                                   ||
||  Duration            |  31536000                              ||
||  End                 |  2023-08-27T13:29:44+00:00             ||
||  FixedPrice          |  59.0                                  ||
||  InstanceCount       |  10                                    ||
||  InstanceTenancy     |  default                               ||
||  InstanceType        |  t3.micro                              ||
||  OfferingClass       |  standard                              ||
||  OfferingType        |  All Upfront                           ||
```

```
||  ProductDescription  |  Linux/UNIX                                ||
||  ReservedInstancesId |  a1b2c3d4-5678-90ab-cdef-EXAMPLE11111      ||
||  Scope               |  Region                                    ||
||  Start               |  2022-08-27T13:29:45.938000+00:00          ||
||  State               |  active                                    ||
||  UsagePrice          |  0.0                                       ||
|+---------------------+--------------------------------------------+|
|||                      RecurringCharges                           |||
||+----------------------------------+---------------------------+||
|||  Amount                          |  0.0                      |||
|||  Frequency                       |  Hourly                   |||
||+----------------------------------+---------------------------+||
```

PowerShell

Untuk melihat jumlah Instans Terpesan yang telah Anda beli

Gunakan Get-EC2ReservedInstance Cmdlet dan tentukan ID konfigurasi Instance Cadangan.

```
Get-EC2ReservedInstance -ReservedInstancesId a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Berikut ini adalah output contoh. Bid `InstanceCount` ang menunjukkan bahwa ada 10 Instans Cadangan untuk konfigurasi ini.

```
AvailabilityZone    :
CurrencyCode        : USD
Duration            : 31536000
End                 : 1/12/2017 8:57:08 PM
FixedPrice          : 0
InstanceCount       : 10
InstanceTenancy     : default
InstanceType        : t3.medium
OfferingClass       : standard
OfferingType        : All Upfront
ProductDescription  : Windows
RecurringCharges    : {}
ReservedInstancesId : a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
Scope               : Region
Start               : 10/12/2016 4:00:00 PM
State               : active
Tags                : {}
UsagePrice          : 0
```

Pertimbangan-pertimbangan

Instans Cadangan regional menerapkan diskon untuk Inst On-Demand ans yang sedang berjalan. Batas On-Demand instans default adalah 20. Anda tidak dapat melebihi batas On-Demand Instans yang sedang berjalan dengan membeli Instans Cadangan regional. Misalnya, jika Anda sudah memiliki 20 On-Demand Instans yang sedang berjalan, dan Anda membeli 20 Instans Cadangan regional, 20 Instans Cadangan regional digunakan untuk menerapkan diskon pada 20 On-Demand Instans yang sedang berjalan. Jika Anda membeli lebih banyak Instans Cadangan regional, Anda tidak akan dapat meluncurkan lebih banyak instans karena Anda telah mencapai batas On-Demand Instans.

Sebelum membeli Instans Cadangan regional, pastikan batas On-Demand Instans Anda cocok atau melebihi jumlah Instans Cadangan regional yang ingin Anda miliki. Jika diperlukan, pastikan Anda meminta peningkatan batas Inst On-Demand ans sebelum membeli Instans Cadangan regional lainnya.

Instans Terpesan zonal—Instans Terpesanyang dibeli untuk Zona Ketersediaan tertentu—memberikan reservasi kapasitas serta diskon. Anda dapat melampaui batas On-Demand Instans yang sedang berjalan dengan membeli Instans Cadangan zona. Misalnya, jika Anda sudah memiliki 20 On-Demand Instans yang sedang berjalan, dan Anda membeli 20 Instans Cadangan zona, Anda dapat meluncurkan 20 On-Demand Instans lagi yang sesuai dengan spesifikasi Instans Cadangan zona Anda, sehingga Anda mendapatkan total 40 instans yang sedang berjalan.

Melihat kuota Instans Terpesan Anda dan meminta peningkatan kuota

Konsol Amazon EC2 menyediakan informasi kuota. Anda juga dapat meminta peningkatan kuota. Lihat informasi yang lebih lengkap di Melihat kuota Anda saat ini dan Meminta peningkatan.

## Instans Spot

Instans Spot adalah instance yang menggunakan kapasitas EC2 cadangan yang tersedia dengan harga kurang dari On-Demand harga. Karena Instans Spot memungkinkan Anda meminta instans EC2 yang tidak digunakan dengan diskon besar, Anda dapat menurunkan biaya Amazon EC2 secara signifikan. Harga per jam untuk Instans Spot disebut harga Spot. Harga Spot dari setiap tipe instans di setiap Zona Ketersediaan ditetapkan oleh Amazon EC2, dan disesuaikan secara bertahap berdasarkan pasokan dan permintaan jangka panjang untuk Instans Spot. Instans Spot Anda berjalan setiap kali kapasitas tersedia.

Instans Spot adalah pilihan hemat biaya jika Anda dapat bersikap fleksibel tentang kapan aplikasi Anda berjalan dan apakah aplikasi Anda dapat diinterupsi. Misalnya, Instans Spot sangat cocok

untuk analisis data, pekerjaan batch, pemrosesan latar belakang, dan tugas opsional. Untuk informasi selengkapnya, lihat Instans Spot Amazon EC2.

Untuk perbandingan opsi pembelian yang berbeda untuk instans EC2, lihat Opsi penagihan dan pembelian Amazon EC2.

## Konsep

Sebelum memulai Instans Spot, Anda harus terbiasa dengan konsep berikut:

- Kolam kapasitas spot – Satu set instans EC2 yang tidak digunakan dengan tipe instans yang sama (misalnya, `m5.large`) dan Zona Ketersediaan yang sama pula.
- Harga spot – Harga Instans Spot saat ini per jam.
- Permintaan Instans Spot – Meminta Instans Spot. Saat kapasitas tersedia, Amazon EC2 memenuhi permintaan Anda. Permintaan Instans Spot bisa satu kali atau tetap. Amazon EC2 secara otomatis mengirimkan kembali permintaan Instans Spot yang persisten setelah Instans Spot yang terkait dengan permintaan tersebut diinterupsi.
- Rekomendasi penyeimbangan kembali instans EC2 – Amazon EC2 memancarkan sinyal rekomendasi penyeimbangan kembali instans untuk memberi tahu Anda bahwa Instans Spot berisiko tinggi mengalami interupsi. Sinyal ini memberikan kesempatan untuk secara proaktif menyeimbangkan kembali beban kerja Anda di Instans Spot yang ada atau yang baru tanpa harus menunggu pemberitahuan interupsi Instans Spot selama dua menit.
- Interupsi Instans Spot – Amazon EC2 mengakhiri, menghentikan, atau menghibernasi Instans Spot Anda saat Amazon EC2 membutuhkan kapasitas kembali. Amazon EC2 memberikan pemberitahuan interupsi Instans Spot, yang memberikan peringatan dua menit pada instans sebelum diinterupsi.

## Perbedaan antara Instans Spot dan On-Demand Instans

Tabel berikut mencantumkan perbedaan utama antara Instans Spot dan On-Demand Instans.

| | Instans Spot | On-Demand Contoh |
|---|---|---|
| Waktu peluncuran | Hanya dapat diluncurkan segera jika permintaan Instans Spot dan kapasitas tersedia. | Hanya dapat diluncurkan segera jika Anda membuat permintaan peluncuran manual dan kapasitas tersedia. |

| | Instans Spot | On-Demand Contoh |
|---|---|---|
| Kapasitas yang tersedia | Jika kapasitas tidak tersedia, permintaan Instans Spot akan terus membuat permintaan peluncuran secara otomatis hingga kapasitas tersedia. | Jika kapasitas tidak tersedia saat Anda membuat permintaan peluncuran, Anda mendapatkan pesan kesalahan kapasitas tidak mencukupi (ICE). |
| Harga per jam | Harga per jam untuk Instans Spot bervariasi berdasarkan pasokan dan permintaan jangka panjang. | Harga per jam untuk On-Demand Instans bersifat statis. |
| Rekomendasi penyeimbangan kembali | Sinyal yang dipancarkan oleh Amazon EC2 untuk Instans Spot yang sedang berjalan saat instans berada pada risiko interupsi yang tinggi. | Anda menentukan kapan On-Demand Instance terganggu (dihentikan, hibernasi, atau dihentikan). |
| Interupsi instans | Anda dapat menghentikan dan memulai Instans Amazon EBS-backed Spot. Selain itu, Amazon EC2 dapat menginterupsi Instans Spot individu jika kapasitas tidak lagi tersedia. | Anda menentukan kapan On-Demand Instance terganggu (dihentikan, hibernasi, atau dihentikan). |

## Penetapan harga dan penghematan

Anda membayar harga Spot untuk Instans Spot, yang ditetapkan oleh Amazon EC2 dan disesuaikan secara bertahap berdasarkan pasokan dan permintaan jangka panjang untuk Instans Spot. Instans Spot berjalan hingga Anda mengakhirinya, kapasitas tidak lagi tersedia, atau grup Amazon EC2 Auto Scaling Anda mengakhirinya selama diskalakan ke dalam.

Jika Anda atau Amazon EC2 menginterupsi Instans Spot yang sedang berjalan, Anda akan dikenai biaya untuk detik yang digunakan atau satu jam penuh, atau Anda tidak dikenai biaya, tergantung sistem operasi yang digunakan dan siapa yang menginterupsi Instans Spot. Untuk informasi selengkapnya, lihat Penagihan untuk Instans Spot yang diinterupsi.

Instans Spot tidak tercakup dalam Paket Tabungan. Jika Anda memiliki Paket Tabungan, itu tidak memberikan penghematan tambahan selain penghematan yang sudah Anda dapatkan dari

menggunakan Instans Spot. Selain itu, pengeluaran Anda untuk Instans Spot tidak menerapkan komitmen dalam Paket Tabungan Komputasi Anda.

Tampilkan harga

Untuk melihat harga Spot terendah saat ini (diperbarui setiap lima menit) per Wilayah AWS dan jenis instans, lihat halaman Harga Instans Spot Amazon EC2.

Untuk melihat riwayat harga Spot selama tiga bulan terakhir, gunakan konsol Amazon EC2 atau perintah describe- spot-price-history. Untuk informasi selengkapnya, lihat Lihat riwayat harga Instans Spot.

Kami secara independen memetakan Zona Ketersediaan ke kode untuk masing-masing Akun AWS. Oleh karena itu, Anda bisa mendapatkan hasil yang berbeda untuk kode Zona Ketersediaan yang sama (misalnya, `us-west-2a`) di antara akun yang berbeda.

Tampilkan penghematan

Anda dapat menampilkan penghematan yang dihasilkan dari penggunaan Instans Spot untuk satu Armada Spot atau untuk semua Instans Spot. Anda dapat menampilkan penghematan yang dilakukan dalam satu jam terakhir atau tiga hari terakhir, dan Anda dapat menampilkan biaya rata-rata per jam vCPU dan per jam memori (GiB). Penghematan diperkirakan dan mungkin berbeda dari penghematan sebenarnya karena tidak menyertakan penyesuaian penagihan untuk penggunaan Anda. Untuk informasi selengkapnya tentang menampilkan informasi penghematan, lihat Penghematan dari pembelian Instans Spot.

Tampilkan penagihan

Tagihan Anda memberikan detail tentang penggunaan layanan Anda. Untuk informasi selengkapnya, lihat Melihat tagihan Anda dalam Panduan Pengguna AWS Billing .

## Praktik terbaik untuk Amazon EC2 Spot

Amazon EC2 menyediakan akses ke kapasitas komputasi EC2 cadangan di Instans Spot AWS Cloud melalui dengan penghematan hingga 90% dibandingkan dengan harga. On-Demand Satu-satunya perbedaan antara On-Demand Instans dan Instans Spot adalah Instans Spot dapat terganggu oleh Amazon EC2, dengan pemberitahuan dua menit, jika Amazon EC2 perlu mendapatkan kembali kapasitas. Untuk memastikan pengalaman terbaik dengan Instans Spot, penting untuk memahami dan menerapkan praktik terbaik untuk penggunaannya.

Instans Spot direkomendasikan untuk aplikasi tanpa stateless, toleransi kesalahan, dan fleksibel. Misalnya, Instans Spot bekerja dengan baik untuk data besar, beban kerja kontainer, server web tanpa status CI/CD, komputasi kinerja tinggi (HPC), dan beban kerja rendering.

Saat berjalan, Instans Spot persis sama dengan On-Demand Instans. Namun, Spot tidak menjamin bahwa Anda dapat mempertahankan instans agar berjalan cukup lama untuk menyelesaikan beban kerja Anda. Spot juga tidak menjamin bahwa Anda bisa langsung mendapatkan ketersediaan instans yang Anda cari, atau bahwa Anda selalu bisa mendapatkan kapasitas agregat yang Anda minta. Selain itu, interupsi dan kapasitas Instans Spot dapat berubah dari waktu ke waktu karena ketersediaan Instans Spot bervariasi berdasarkan pasokan dan permintaan. Selain itu, performa masa lalu bukanlah jaminan untuk hasil di masa mendatang.

Instans Spot tidak cocok untuk beban kerja yang tidak fleksibel, stateful, tidak toleran terhadap kesalahan, atau digabungkan erat di antara simpul instans. Kami tidak merekomendasikan Instans Spot untuk beban kerja yang tidak toleran terhadap periode sesekali ketika seluruh kapasitas target tidak sepenuhnya tersedia. Meskipun mengikuti praktik terbaik Spot untuk bersikap fleksibel mengenai jenis instans dan Zona Ketersediaan memberikan peluang terbaik untuk ketersediaan tinggi, tidak ada jaminan bahwa kapasitas akan tersedia, karena lonjakan permintaan On-Demand Instans dapat mengganggu beban kerja pada Instans Spot.

Kami sangat tidak menyarankan menggunakan Instans Spot untuk beban kerja ini atau mencoba untuk gagal beralih ke On-Demand Instans untuk menangani interupsi atau periode ketidaktersediaan. Gagal beralih ke On-Demand Instans dapat secara tidak sengaja menyebabkan interupsi untuk Instans Spot lainnya. Selain itu, jika Instans Spot untuk kombinasi tipe instans dan Availability Zone terganggu, mungkin akan sulit bagi Anda untuk mendapatkan On-Demand Instans dengan kombinasi yang sama.

Terlepas dari apakah Anda pengguna Spot berpengalaman atau baru menggunakan Instans Spot, jika saat ini Anda mengalami masalah terkait interupsi atau ketersediaan Instans Spot, kami sarankan Anda mengikuti praktik terbaik ini untuk mendapatkan pengalaman terbaik menggunakan layanan Spot.

Praktik terbaik Spot

- Menyiapkan instans individu untuk interupsi
- Bersikaplah fleksibel terkait tipe instans dan Zona Ketersediaan
- Gunakan pemilihan tipe instance berbasis atribut
- Gunakan skor penempatan Spot untuk mengidentifikasi Wilayah dan Zona Ketersediaan yang optimal

- Menggunakan grup EC2 Auto Scaling atau EC2 Fleet untuk mengelola kapasitas agregat Anda
- Menggunakan strategi alokasi harga dan kapasitas yang dioptimalkan
- Gunakan terintegrasi AWS layanan untuk mengelola Instans Spot Anda
- Metode permintaan Spot mana yang terbaik untuk digunakan?

Menyiapkan instans individu untuk interupsi

Cara terbaik agar Anda dapat menangani interupsi Instans Spot dengan baik adalah dengan merancang aplikasi Anda agar toleran terhadap kesalahan. Untuk melakukannya, Anda dapat memanfaatkan rekomendasi penyeimbangan kembali instans EC2 dan notifikasi interupsi Instans Spot.

Rekomendasi penyeimbangan kembali Instans EC2 adalah sinyal yang memberi tahu Anda saat Instans Spot berada pada risiko interupsi yang tinggi. Sinyal tersebut memberi Anda kesempatan untuk secara proaktif mengelola Instans Spot sebelum pemberitahuan interupsi Instans Spot dua menit. Anda dapat memutuskan untuk menyeimbangkan kembali beban kerja Anda ke Instans Spot baru atau lama yang tidak berisiko tinggi mengalami interupsi. Kami telah mempermudah Anda untuk menggunakan sinyal ini dengan fitur Penyeimbangan Kembali Kapasitas di grup Auto Scaling dan EC2 Fleet.

Pemberitahuan interupsi Instans Spot adalah peringatan yang dikeluarkan dua menit sebelum Amazon EC2 menginterupsi Instans Spot. Jika beban kerja Anda “fleksibel waktu”, Anda dapat mengonfigurasi Instans Spot untuk dihentikan atau dihibernasi, alih-alih diakhiri, saat terinterupsi. Amazon EC2 secara otomatis menghentikan atau menghibernasi Instans Spot Anda saat terjadi interupsi, dan secara otomatis melanjutkan instans saat kami memiliki kapasitas yang tersedia.

Kami menyarankan Anda membuat aturan di Amazon EventBridge yang menangkap rekomendasi penyeimbangan ulang dan pemberitahuan gangguan, lalu memicu titik pemeriksaan untuk kemajuan beban kerja Anda atau menangani gangguan dengan baik. Untuk informasi selengkapnya, lihat Pantau sinyal rekomendasi penyeimbangan kembali. Untuk contoh terperinci yang akan memandu Anda tentang cara membuat dan menggunakan aturan peristiwa, lihat Memanfaatkan Notifikasi Interupsi Instans Spot Amazon EC2.

Untuk informasi lebih lanjut, lihat Rekomendasi penyeimbangan ulang instans EC2 dan.Interupsi Instans Spot

Bersikaplah fleksibel terkait tipe instans dan Zona Ketersediaan

Kolam kapasitas Spot adalah sekumpulan instans EC2 yang tidak digunakan dengan tipe instans yang sama (misalnya, `m5.large`) dan Zona Ketersediaan yang sama pula (misalnya, us-east-1a). Anda harus fleksibel terkait tipe instans yang Anda minta dan di Zona Ketersediaan mana Anda dapat menerapkan beban kerja. Hal ini memberi Spot peluang yang lebih baik untuk menemukan dan mengalokasikan jumlah kapasitas komputasi yang Anda butuhkan. Misalnya, jangan hanya meminta `c5.large` jika Anda ingin menggunakan keluarga c4, m5, dan m4 yang lebih besar.

Tergantung kebutuhan tertentu, Anda dapat mengevaluasi tipe instans yang bisa digunakan secara fleksibel untuk memenuhi persyaratan komputasi Anda. Jika beban kerja dapat diskalakan secara vertikal, Anda harus menyertakan tipe instans yang lebih besar (lebih banyak vCPU dan memori) dalam permintaan Anda. Jika Anda hanya dapat menskalakan secara horizontal, Anda harus menyertakan jenis instans generasi lama karena kurang diminati On-Demand pelanggan.

Aturan praktis yang baik adalah bersikap fleksibel pada setidaknya 10 tipe instans untuk setiap beban kerja. Selain itu, pastikan semua Zona Ketersediaan dikonfigurasi untuk digunakan di VPC Anda dan dipilih untuk beban kerja Anda.

Gunakan pemilihan tipe instance berbasis atribut

Dengan pemilihan jenis instans berbasis atribut, Anda dapat menentukan atribut instans—seperti vCPU, memori, dan penyimpanan—untuk beban kerja yang ingin Anda jalankan. EC2 Auto Scaling atau EC2 Fleet kemudian akan secara otomatis mengidentifikasi dan meluncurkan instans yang sesuai dengan atribut yang Anda tentukan. Ini menghilangkan upaya yang diperlukan untuk memilih jenis instans tertentu secara manual, yang memerlukan pemahaman mendalam tentang penawaran setiap jenis instans.

Selain itu, pemilihan tipe instance berbasis atribut memungkinkan Anda menggunakan jenis instans yang baru dirilis secara otomatis saat tersedia. Ini memastikan akses tanpa batas ke jangkauan kapasitas Instans Spot yang semakin luas.

Attribute-based pemilihan tipe instance sangat ideal untuk beban kerja dan kerangka kerja yang dapat fleksibel tentang jenis instans yang dijalankannya, seperti High Performance Computing (HPC) dan beban kerja data besar.

Untuk informasi selengkapnya, lihat Membuat grup instans campuran menggunakan pemilihan jenis instans berbasis atribut di Panduan Pengguna Penskalaan Otomatis Amazon EC2 dan Tentukan atribut untuk pemilihan tipe instans untuk EC2 Fleet atau Spot Fleet dalam panduan ini.

### Gunakan skor penempatan Spot untuk mengidentifikasi Wilayah dan Zona Ketersediaan yang optimal

Instans Spot adalah kapasitas EC2 yang tidak digunakan, dan kapasitas ini berfluktuasi berdasarkan penawaran dan permintaan EC2. Akibatnya, Anda mungkin tidak selalu mendapatkan kapasitas Spot yang tepat yang Anda butuhkan di lokasi tertentu pada waktu tertentu. Untuk mengurangi ketidakpastian ini, Anda dapat menggunakan fitur skor penempatan Spot. Fitur ini memberikan rekomendasi untuk Wilayah atau Zona Ketersediaan yang lebih mungkin memiliki kapasitas yang cukup untuk memenuhi kebutuhan kapasitas Spot Anda tanpa mengharuskan Anda meluncurkan Instans Spot di lokasi tersebut terlebih dahulu.

Skor penempatan spot paling baik digunakan untuk beban kerja yang fleksibel mengenai jenis instans dan Wilayah atau Zona Ketersediaan yang dapat mereka gunakan. Yang perlu Anda lakukan adalah menentukan kapasitas Spot yang Anda butuhkan, persyaratan jenis instans Anda, dan apakah Anda menginginkan rekomendasi untuk Wilayah atau Zona Ketersediaan. Sebagai imbalannya, Anda menerima skor mulai dari 1 hingga 10 untuk setiap Wilayah atau Zona Ketersediaan, yang menunjukkan kemungkinan berhasil menyediakan kapasitas Spot yang Anda minta di lokasi tersebut. Skor 10 menunjukkan bahwa permintaan Spot Anda sangat mungkin berhasil.

Penting untuk dicatat bahwa skor penempatan Spot adalah rekomendasi point-in-time, karena kapasitas dapat bervariasi dari waktu ke waktu. Itu tidak menjamin kapasitas yang tersedia atau memprediksi risiko gangguan.

Anda dapat menggunakan fitur skor penempatan Spot di konsol Amazon EC2, AWS CLI, atau SDK. Untuk informasi selengkapnya, lihat Skor penempatan Spot.

### Menggunakan grup EC2 Auto Scaling atau EC2 Fleet untuk mengelola kapasitas agregat Anda

Spot memungkinkan Anda untuk berpikir dalam hal kapasitas agregat—dalam unit yang mencakup vCPU, memori, penyimpanan, atau throughput jaringan—bukan berpikir dalam hal kapasitas instans. Grup Auto Scaling dan EC2 Fleet memungkinkan Anda untuk meluncurkan dan mempertahankan kapasitas target, serta secara otomatis meminta sumber daya untuk menggantikan sumber daya yang terganggu atau diakhiri secara manual. Saat mengonfigurasi grup Auto Scaling atau EC2 Fleet, Anda hanya perlu menentukan tipe instans dan kapasitas target berdasarkan kebutuhan aplikasi Anda. Untuk informasi selengkapnya, lihat Grup Auto Scaling dalam Panduan Pengguna Amazon EC2 Auto Scaling dan Membuat Armada EC2 dalam panduan pengguna ini.

Menggunakan strategi alokasi harga dan kapasitas yang dioptimalkan

Strategi alokasi dalam grup Auto Scaling membantu Anda menyediakan kapasitas target tanpa perlu mencari kolam kapasitas Spot secara manual dengan kapasitas tak terpakai. Kami merekomendasikan penggunaan strategi `price-capacity-optimized` karena strategi ini secara otomatis menyediakan instans dari kolam kapasitas Spot yang juga memiliki potensi harga paling rendah. Anda juga dapat memanfaatkan strategi alokasi `price-capacity-optimized` di Armada EC2. Karena kapasitas Instans Spot Anda bersumber dari kolam dengan kapasitas optimal, hal ini mengurangi kemungkinan bahwa Instans Spot Anda diklaim kembali. Untuk informasi selengkapnya, lihat Strategi alokasi untuk beberapa jenis instans di Panduan Pengguna Penskalaan Otomatis Amazon EC2 dan Ketika beban kerja memiliki biaya interupsi yang tinggi dalam panduan pengguna ini.

Gunakan terintegrasi AWS layanan untuk mengelola Instans Spot Anda

AWS Layanan lain terintegrasi dengan Spot untuk mengurangi biaya komputasi secara keseluruhan tanpa perlu mengelola instans atau armada individu. Sebaiknya pertimbangkan solusi berikut untuk beban kerja yang berlaku: Amazon EMR, Amazon Elastic Container Service AWS Batch, Amazon Elastic Kubernetes Service, Amazon SageMaker AI, AWS Elastic Beanstalk dan Amazon Servers. GameLift Untuk mempelajari selengkapnya tentang praktik terbaik Spot dengan layanan ini, lihat Situs Web Lokakarya Instans Spot Amazon EC2.

Metode permintaan Spot mana yang terbaik untuk digunakan?

Gunakan tabel berikut untuk menentukan API yang akan digunakan saat meminta Instans Spot.

| API | Kapan harus menggunakan? | Kasus penggunaan | Haruskah saya menggunakan API ini? |
|---|---|---|---|
| CreateAutoScalingGroup | • Anda memerlukan beberapa instans dengan konfigurasi tunggal atau konfigurasi campuran.<br>• | Buat grup Auto Scaling yang mengelola siklus hidup instans Anda sambil mempertahankan jumlah instans yang diinginkan. Mendukung penskalaan horizonta | Ya |

| API | Kapan harus menggunakan? | Kasus penggunaan | Haruskah saya menggunakan API ini? |
| --- | --- | --- | --- |
| | Anda ingin mengotomatiskan manajemen siklus hidup melalui API yang dapat dikonfigurasi. | l (menambahkan lebih banyak instans) antara batas minimum dan maksimum yang ditentukan. | |
| CreateFleet | • Anda memerlukan beberapa instans dengan konfigurasi tunggal atau konfigurasi campuran.<br>• Anda ingin mengelola sendiri siklus hidup instans Anda.<br>• Jika Anda tidak memerlukan penskalaan otomatis, kami sarankan Anda menggunakan armada tipe `instant`. | Buat armada Instans dan On-Demand Instans Spot dalam satu permintaan, dengan beberapa spesifikasi peluncuran yang bervariasi menurut jenis instans, AMI, Zona Ketersediaan, atau subnet. Strategi alokasi Instans Spot default ke `lowest-price` per unit, tetapi Anda dapat mengubahnya menjadi `price-capacity-optimized` , `capacity-optimized` , atau `diversified` . | Ya – dalam mode `instant` jika Anda tidak memerlukan penskalaan otomatis |

| API | Kapan harus menggunakan? | Kasus penggunaan | Haruskah saya menggunakan API ini? |
| --- | --- | --- | --- |
| RunInstances | • Anda sudah menggunakan RunInstances API untuk meluncurkan On-Demand Instans, dan Anda hanya ingin mengubah ke meluncurkan Instans Spot dengan mengubah satu parameter.<br>• Anda tidak memerlukan banyak instans dengan tipe instans yang berbeda. | Luncurkan sejumlah tertentu instans menggunakan AMI dan satu tipe instans. | Tidak - karena RunInstances tidak mengizinkan jenis instance campuran dalam satu permintaan |

| API | Kapan harus menggunakan? | Kasus penggunaan | Haruskah saya menggunakan API ini? |
| --- | --- | --- | --- |
| RequestSpotFleet | • Kami sangat tidak menyarankan menggunakan RequestSpotFleet API karena ini adalah API lama tanpa investasi yang direncanakan.<br>• Jika Anda ingin mengelola siklus hidup instans, gunakan CreateFleet API.<br>• Jika Anda tidak ingin mengelola siklus hidup instans, gunakan CreateAutoScalingGroup API. | JANGAN GUNAKAN. RequestSpotFleet adalah API lama tanpa investasi yang direncanakan. | Tidak |

| API | Kapan harus menggunakan? | Kasus penggunaan | Haruskah saya menggunakan API ini? |
| --- | --- | --- | --- |
| RequestSpotInstances | • Kami sangat tidak menyarankan menggunakan RequestSpotInstances API karena ini adalah API lama tanpa investasi yang direncanakan. | JANGAN GUNAKAN. RequestSpotInstances adalah API lama tanpa investasi yang direncanakan. | Tidak |

## Cara kerja Instans Spot

Untuk meluncurkan Instans Spot, Anda dapat membuat permintaan Instans Spot, atau Amazon EC2 membuat permintaan Instans Spot atas nama Anda. Instans Spot diluncurkan ketika permintaan Instans Spot dipenuhi.

Anda dapat meluncurkan Instans Spot menggunakan beberapa layanan berbeda. Untuk informasi selengkapnya, lihat Memulai Instans Spot Amazon EC2. Dalam panduan pengguna ini, kami menjelaskan cara-cara berikut untuk meluncurkan Instans Spot menggunakan EC2:

- Anda dapat membuat permintaan Instans Spot dengan menggunakan wizard instance peluncuran di konsol Amazon EC2 atau perintah run- instance. Untuk informasi selengkapnya, lihat Mengelola Instans Spot Anda.
- Anda dapat membuat EC2 Fleet, tempat Anda menentukan jumlah Instans Spot yang diinginkan. Amazon EC2 membuat permintaan Instans Spot atas nama Anda untuk setiap Instans Spot yang ditentukan di EC2 Fleet. Untuk informasi selengkapnya, lihat Membuat Armada EC2.
- Anda dapat membuat permintaan Armada Spot, tempat Anda menentukan jumlah Instans Spot yang diinginkan. Amazon EC2 membuat permintaan Instans Spot atas nama Anda untuk setiap Instans Spot yang ditentukan pada permintaan Armada Spot. Untuk informasi selengkapnya, lihat Membuat Armada Spot.

> **Note**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Instans Spot Anda diluncurkan jika ada kapasitas yang tersedia. Instans Spot berjalan hingga Anda menghentikan atau mengakhirinya, atau hingga Amazon EC2 menginterupsi (dikenal sebagai Interupsi Instans Spot). Amazon EC2 dapat menghentikan, menghentikan, atau melakukan hibernasi Instans Spot saat menginterupsinya.

Saat menggunakan Instans Spot, Anda harus siap menghadapi interupsi. Amazon EC2 dapat menginterupsi Instans Spot Anda saat permintaan Instans Spot naik atau saat pasokan Instans Spot berkurang. Saat menginterupsi Instans Spot, Amazon EC2 memberikan pemberitahuan interupsi Instans Spot, yang memberi instans peringatan dua menit sebelum Amazon EC2 menginterupsi. Anda tidak dapat mengaktifkan perlindungan pengakhiran untuk Instans Spot. Untuk informasi selengkapnya, lihat Interupsi Instans Spot.

Daftar Isi


- Status permintaan Instans Spot
- Meluncurkan Instans Spot dalam grup peluncuran
- Meluncurkan Instans Spot dalam grup Zona Ketersediaan
- Meluncurkan Instans Spot di VPC
- Luncurkan instans kinerja yang meledak
- Peluncuran pada perangkat keras penyewa tunggal

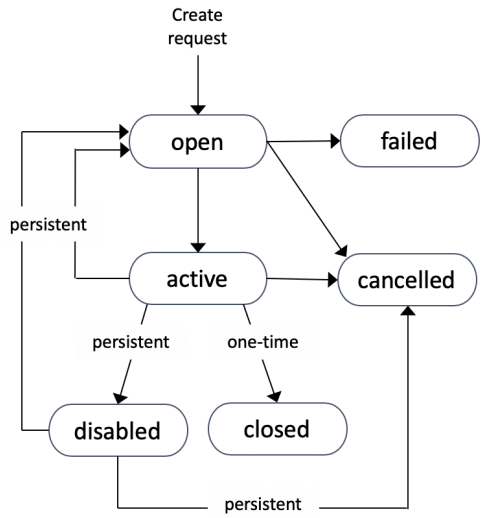

## Status permintaan Instans Spot

Permintaan Instans Spot dapat berada dalam salah satu status berikut:

- `open` – Permintaan menunggu untuk dipenuhi.
- `active` – Permintaan Instans Spot terpenuhi dan memiliki Instans Spot terkait.
- `failed` – Permintaan memiliki satu atau beberapa parameter buruk.
- `closed` – Instans Spot diinterupsi atau diakhiri.

- `disabled` – Anda menghentikan Instans Spot.
- `cancelled` – Anda membatalkan permintaan, atau permintaan kedaluwarsa.

Ilustrasi berikut mewakili transisi antarstatus permintaan status. Perhatikan bahwa transisi bergantung pada tipe permintaan (satu kali atau tetap).




Permintaan Instans Spot satu kali tetap aktif hingga Amazon EC2 meluncurkan Instans Spot, permintaan kedaluwarsa, atau Anda membatalkan permintaan. Jika kapasitas tidak tersedia, Instans Spot Anda diakhiri dan permintaan Instans Spot ditutup.

Permintaan Instans Spot persisten tetap aktif hingga kedaluwarsa atau Anda membatalkannya, bahkan jika permintaan dipenuhi. Jika kapasitas tidak tersedia, Instans Spot Anda diinterupsi. Setelah instans Anda diinterupsi, saat kembali kapasitas tersedia, Instans Spot akan dimulai jika dihentikan atau dilanjutkan jika hibernasi. Anda dapat menghentikan Instans Spot dan memulainya lagi jika kapasitas tersedia. Jika Instans Spot diakhiri (terlepas dari apakah Instans Spot dalam status berhenti atau berjalan), permintaan Instans Spot dibuka kembali dan Amazon EC2 meluncurkan Instans Spot baru. Untuk informasi selengkapnya, lihat Menghentikan Instans Spot, Memulai Instans Spot, dan Menghentikan Instans Spot.

Anda dapat melacak status permintaan Instans Spot, serta status Instans Spot yang diluncurkan, melalui status. Untuk informasi selengkapnya, lihat Mendapatkan status permintaan Instance Spot.

Meluncurkan Instans Spot dalam grup peluncuran

Tentukan grup peluncuran dalam permintaan Instans Spot Anda untuk memberi tahu Amazon EC2 agar meluncurkan sekumpulan Instans Spot hanya jika dapat meluncurkan semuanya. Selain itu,

jika layanan Spot harus mengakhiri salah satu instans dalam grup peluncuran, layanan tersebut harus mengakhiri semuanya. Namun, jika Anda mengakhiri satu atau beberapa instans dalam grup peluncuran, Amazon EC2 tidak mengakhiri instans yang tersisa di grup peluncuran.

Meskipun opsi ini dapat berguna, menambahkan batasan ini dapat mengurangi kemungkinan permintaan Instans Spot Anda dipenuhi dan meningkatkan kemungkinan Instans Spot Anda diakhiri. Misalnya, grup peluncuran Anda menyertakan instans di beberapa Zona Ketersediaan. Jika kapasitas di salah satu Zona Ketersediaan ini menurun dan tidak lagi tersedia, Amazon EC2 akan mengakhiri semua instans untuk grup peluncuran.

Jika Anda membuat permintaan Instans Spot sukses lain yang menetapkan grup peluncuran yang sama (yang ada) sebagai permintaan sukses sebelumnya, maka instans baru akan ditambahkan ke grup peluncuran. Selanjutnya, jika sebuah instans dalam grup peluncuran ini diakhiri, semua instans dalam grup peluncuran akan diakhiri, yang mencakup instans yang diluncurkan oleh permintaan pertama dan kedua.

Meluncurkan Instans Spot dalam grup Zona Ketersediaan

Tentukan grup Zona Ketersediaan dalam permintaan Instans Spot Anda untuk memberi tahu Amazon EC2 agar meluncurkan sekumpulan Instans Spot di Zona Ketersediaan yang sama. Amazon EC2 tidak perlu menginterupsi semua instans dalam grup Zona Ketersediaan pada saat yang bersamaan. Jika Amazon EC2 harus menginterupsi salah satu instans dalam grup Zona Ketersediaan, instans yang lainnya tetap berjalan.

Meskipun opsi ini dapat berguna, menambahkan batasan ini dapat menurunkan kemungkinan permintaan Instans Spot Anda dipenuhi.

Jika Anda menentukan grup Zona Ketersediaan, tetapi tidak menentukan Zona Ketersediaan dalam permintaan Instans Spot, hasilnya bergantung pada jaringan yang Anda tentukan.

VPC default

Amazon EC2 menggunakan Zona Ketersediaan untuk subnet yang ditentukan. Subnet yang tidak Anda tentukan akan memilih Zona Ketersediaan dan subnet default-nya, tetapi belum tentu zona harga terendah. Jika Anda menghapus subnet default untuk Zona Ketersediaan, Anda harus menentukan subnet yang berbeda.

VPC Non-default

Amazon EC2 menggunakan Zona Ketersediaan untuk subnet yang ditentukan.

### Meluncurkan Instans Spot di VPC

Anda menentukan subnet untuk Instans Spot dengan cara yang sama seperti Anda menentukan subnet untuk Instans Anda On-Demand .

- [VPC Default] Jika ingin Instans Spot diluncurkan di Zona Ketersediaan dengan harga rendah tertentu, Anda harus menentukan subnet yang sesuai dalam permintaan Instans Spot Anda. Jika Anda tidak menentukan subnet, Amazon EC2 akan memilihnya untuk Anda, dan Zona Ketersediaan untuk subnet ini mungkin tidak memiliki harga Spot terendah.
- [VPC Non-default] Anda harus menentukan subnet untuk Instans Spot Anda.

### Luncurkan instans kinerja yang meledak

Tipe instans T adalah instans performa yang dapat melonjak. Jika Anda meluncurkan Instans Spot menggunakan tipe instans performa yang dapat melonjak, dan jika Anda berencana untuk segera menggunakan Instans Spot performa dapat melonjak dan untuk durasi yang singkat, tanpa waktu idle untuk mengakumulasi kredit CPU, kami menyarankan Anda untuk meluncurkannya dalam mode Standar agar tidak membayar biaya yang lebih tinggi. Jika Anda meluncurkan Instans Spot performa yang dapat melonjak dalam Mode tak terbatas dan langsung melonjakkan CPU, Anda akan menghabiskan kredit surplus untuk lonjakan. Jika Anda menggunakan instans untuk durasi yang singkat, instans tersebut tidak memiliki waktu untuk mengkumulasi kredit CPU untuk membayar kredit surplus, dan Anda akan dikenai biaya untuk kredit surplus saat Anda mengakhiri instans.

Mode tidak terbatas cocok untuk Instans Spot dengan performa yang dapat melonjak hanya jika instans tersebut berjalan cukup lama untuk mengakumlasi kredit CPU untuk lonjakan. Jika tidak, pembayaran kredit surplus membuat Instans Spot performa yang dapat melonjak lebih mahal daripada menggunakan instans lain. Untuk informasi selengkapnya, lihat Kapan menggunakan mode tak terbatas versus CPU tetap.

Instans T2, ketika dikonfigurasi dalam mode Standar, dapatkan kredit peluncuran. Instans T2 adalah satu-satunya instans performa yang dapat melonjak yang mendapatkan kredit peluncuran. Kredit peluncuran dimaksudkan untuk memberikan pengalaman peluncuran awal yang produktif untuk instans T2 dengan menyediakan sumber daya komputasi yang memadai untuk mengonfigurasi instans. Peluncuran berulang dari instans T2 untuk mengakses kredit peluncuran baru tidak diizinkan. Jika Anda memerlukan CPU berkelanjutan, Anda dapat memperoleh kredit (dengan berhenti selama beberapa periode), menggunakan mode Tak Terbatas untuk Instans Spot T2, atau menggunakan tipe instans dengan CPU khusus.

Peluncuran pada perangkat keras penyewa tunggal

Anda dapat menjalankan Instans Spot pada perangkat keras penghuni tunggal. Instans Spot Khusus secara fisik terisolasi dari instans milik AWS akun lain. Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2 dan Inst ans Khusus Amazon EC2.

Untuk menjalankan Instans Spot Khusus, lakukan salah satu hal berikut:

- Tentukan penghunian `dedicated` saat Anda membuat permintaan Instans Spot. Untuk informasi selengkapnya, lihat Mengelola Instans Spot Anda.
- Minta Instans Spot di VPC dengan penghunian instans `dedicated`. Untuk informasi selengkapnya, lihat Luncurkan Instans Khusus ke VPC dengan sewa default. Anda tidak dapat meminta Instans Spot dengan sewa `default` jika Anda memintanya di VPC dengan sewa instans sebesar. `dedicated`

Semua keluarga instans mendukung Instans Spot Khusus, kecuali instans T. Untuk setiap keluarga instans yang didukung, hanya ukuran instans atau ukuran metal terbesar yang mendukung Instans Spot Khusus.

## Lihat riwayat harga Instans Spot

Harga Instans Spot ditetapkan oleh Amazon EC2 dan menyesuaikan secara bertahap berdasarkan tren jangka panjang dalam pasokan dan permintaan untuk kapasitas Instans Spot.

Ketika permintaan Spot Anda terpenuhi, Instans Spot Anda diluncurkan dengan harga Spot saat ini, tidak melebihi On-Demand harga. Anda dapat melihat riwayat harga Spot selama 90 hari terakhir, memfilter menurut tipe instans, sistem operasi, dan Zona Ketersediaan.

Untuk harga Instans Spot saat ini, lihat Harga Instans Spot Amazon EC2.

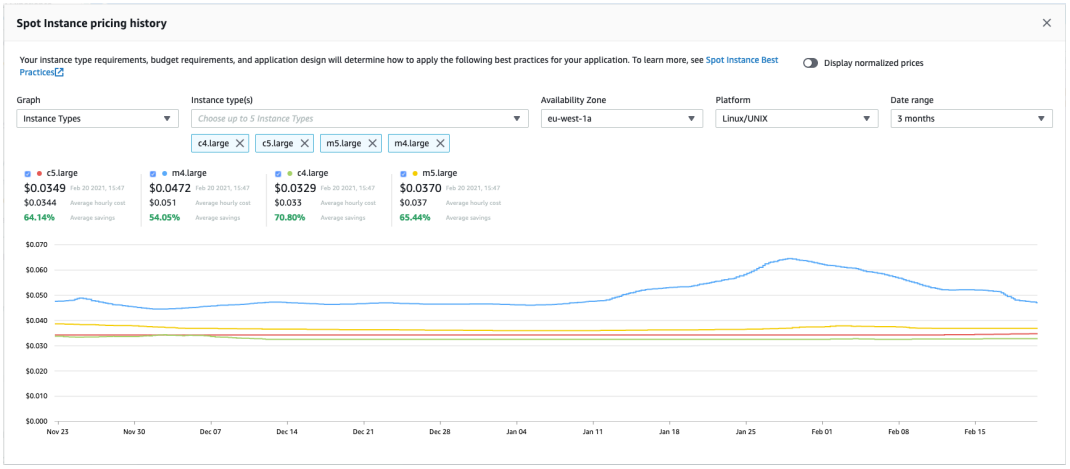
Console

Untuk melihat riwayat harga Spot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih Riwayat harga.
4. Untuk Grafik, pilih untuk membandingkan riwayat harga berdasarkan Zona Ketersediaan atau berdasarkan Tipe Instans.

- Jika Anda memilih Zona Ketersediaan, maka pilih Tipe Instans, sistem operasi (Platform), dan Rentang tanggal untuk melihat riwayat harga.
- Jika Anda memilih Tipe Instans, maka pilih sampai lima Tipe Instans, Zona Ketersediaan, sistem operasi (Platform), dan Rentang tanggal untuk melihat riwayat harga.

Tangkapan layar berikut menunjukkan perbandingan harga untuk tipe instans yang berbeda.




5. Arahkan kursor ke grafik untuk menampilkan harga pada waktu tertentu dalam rentang tanggal yang dipilih. Harga ditampilkan di blok informasi di atas grafik. Harga yang ditampilkan di baris atas menunjukkan harga pada tanggal tertentu. Harga yang ditampilkan di baris kedua menunjukkan harga rata-rata selama rentang tanggal yang dipilih.
6. Untuk menampilkan harga per vCPU, aktifkan Tampilkan harga yang dinormalisasi. Untuk menampilkan harga untuk tipe instans, nonaktifkan Tampilkan harga yang dinormalisasi.

## AWS CLI

Untuk melihat riwayat harga Spot

Gunakan perintah describe-spot-price-history berikut ini.

```
aws ec2 describe-spot-price-history \
    --instance-types c6i.xlarge \
    --product-descriptions "Linux/UNIX" \
    --start-time 2025-04-01T00:00:00 \
    --end-time 2025-04-02T00:00:0
```

PowerShell

Untuk melihat riwayat harga Spot

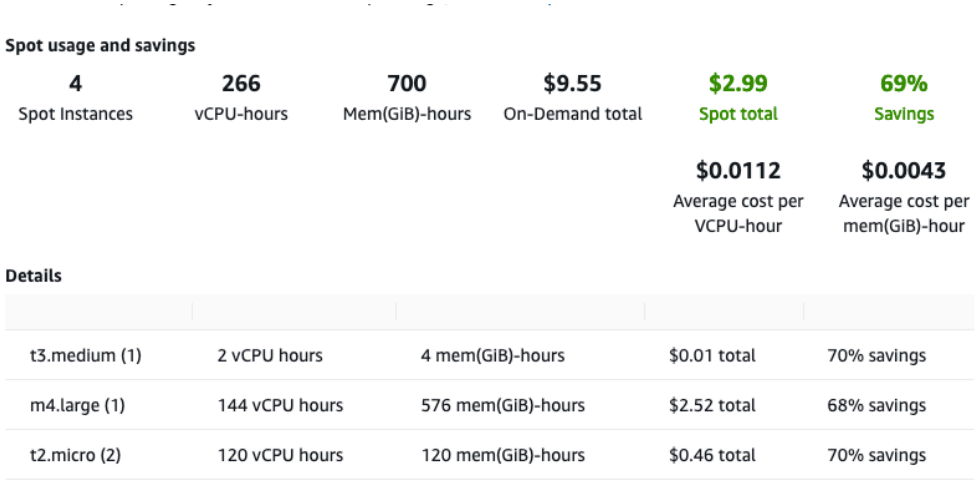
Gunakan Get-EC2SpotPriceHistory cmdlet.

```
Get-EC2SpotPriceHistory `
    -InstanceType c6i.xlarge `
    -ProductDescription "Linux/UNIX" `
    -UtcStartTime 2025-04-01T00:00:00 `
    -UtcEndTime 2025-04-02T00:00:0
```

## Penghematan dari pembelian Instans Spot

Anda dapat melihat informasi penggunaan dan penghematan untuk Instans Spot di tingkat per armada, atau untuk semua Instans Spot yang sedang berjalan. Pada tingkat per armada, informasi penggunaan dan penghematan mencakup semua instans yang diluncurkan dan diakhiri oleh armada. Anda dapat melihat informasi ini dari satu jam terakhir atau tiga hari terakhir.

Tangkapan layar dari bagian Penghematan berikut menunjukkan penggunaan Spot dan informasi penghematan untuk Armada Spot.



**Spot usage and savings**

| 4 | 266 | 700 | $9.55 | $2.99 | 69% |
|---|---|---|---|---|---|
| Spot Instances | vCPU-hours | Mem(GiB)-hours | On-Demand total | Spot total | Savings |
| | | | | $0.0112 | $0.0043 |
| | | | | Average cost per VCPU-hour | Average cost per mem(GiB)-hour |

**Details**

| | | | | |
|---|---|---|---|---|
| t3.medium (1) | 2 vCPU hours | 4 mem(GiB)-hours | $0.01 total | 70% savings |
| m4.large (1) | 144 vCPU hours | 576 mem(GiB)-hours | $2.52 total | 68% savings |
| t2.micro (2) | 120 vCPU hours | 120 mem(GiB)-hours | $0.46 total | 70% savings |

Anda dapat melihat informasi penggunaan dan penghematan berikut:

- Instans Spot – Jumlah Instans Spot yang diluncurkan dan diakhiri oleh Armada Spot. Saat melihat ringkasan penghematan, angka tersebut mewakili semua Instans Spot Anda yang sedang berjalan.
- v CPU-hours — Jumlah jam vCPU yang digunakan di semua Instans Spot untuk kerangka waktu yang dipilih.

- Mem(GiB)-jam – Jumlah jam GiB yang digunakan di semua Instans Spot untuk kerangka waktu yang dipilih.
- On-Demand total — Jumlah total yang akan Anda bayarkan untuk jangka waktu yang dipilih jika Anda meluncurkan instans ini sebagai On-Demand Instans.
- Total Spot – Jumlah total yang harus dibayar untuk kerangka waktu yang dipilih.
- Tabungan — Persentase yang Anda hemat dengan tidak membayar On-Demand harganya.
- Biaya rata-rata per v CPU-hour — Biaya rata-rata per jam penggunaan vCPU di semua Instans Spot untuk jangka waktu yang dipilih, dihitung sebagai berikut: Biaya rata-rata per v CPU-hour = Total Spot/v CPU-hours.
- Biaya rata-rata per mem (GiB) -jam — Biaya rata-rata per jam penggunaan GiBs di semua Instans Spot untuk kerangka waktu yang dipilih, dihitung sebagai berikut: Biaya rata-rata per mem (GiB) -jam = Total Spot/Mem (GiB) -jam.
- Tabel detail – Tipe instans yang berbeda (jumlah instans per tipe instans ada dalam kurung) yang mencakup Armada Spot. Saat melihat ringkasan penghematan, ini mencakup semua Instans Spot Anda yang sedang berjalan.

Informasi penghematan hanya dapat dilihat menggunakan konsol Amazon EC2.

> **Note**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Untuk melihat informasi penghematan untuk Armada Spot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih ID permintaan Armada Spot dan gulir ke bagian Penghematan.

   Atau, pilih kotak centang di sebelah ID permintaan Spot Fleet dan pilih tab Tabungan.
4. Secara default, halaman menampilkan informasi penggunaan dan penghematan selama tiga hari terakhir. Anda dapat memilih satu jam terakhir atau tiga hari terakhir. Untuk Armada Spot yang diluncurkan kurang dari satu jam yang lalu, halaman tersebut menunjukkan perkiraan penghematan untuk satu jam.

Untuk melihat informasi penghematan untuk semua Instans Spot yang sedang berjalan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih Riwayat Penghematan.

## Membuat permintaan Instans Spot

Untuk menggunakan Instans Spot, Anda membuat permintaan Instans Spot yang menyertakan jumlah instans yang diinginkan, tipe instans, dan Zona Ketersediaan. Saat kapasitas tersedia, Amazon EC2 segera memenuhi permintaan Anda. Jika tidak, Amazon EC2 akan menunggu hingga permintaan Anda dipenuhi atau hingga Anda membatalkan permintaan.

Anda dapat menggunakan wizard instance peluncuran di konsol Amazon EC2 atau perintah run-instance untuk meminta Instans Spot dengan cara yang sama seperti Anda dapat meluncurkan Instans. On-Demand Metode ini hanya direkomendasikan karena alasan berikut:

- Anda sudah menggunakan wizard instance peluncuran atau perintah run-instance untuk meluncurkan On-Demand Instans, dan Anda hanya ingin mengubah ke meluncurkan Instans Spot dengan mengubah satu parameter.
- Anda tidak memerlukan banyak instans dengan tipe instans yang berbeda.

Metode ini umumnya tidak disarankan untuk meluncurkan Instans Spot karena Anda tidak dapat menentukan beberapa jenis instans, dan Anda tidak dapat meluncurkan Instans Spot dan On-Demand Instans dalam permintaan yang sama. Untuk metode pilihan untuk meluncurkan Instans Spot, termasuk meluncurkan armada yang menyertakan Instans Spot dan On-Demand Instans dengan beberapa jenis instans, lihat Metode permintaan Spot mana yang terbaik untuk digunakan?

Jika Anda meminta banyak Instans Spot sekaligus, Amazon EC2 membuat permintaan Instans Spot terpisah sehingga Anda dapat melacak status setiap permintaan secara terpisah. Untuk informasi selengkapnya tentang melacak permintaan Instans Spot, lihat Mendapatkan status permintaan Instance Spot.

Console

Untuk membuat permintaan Instance Spot

Langkah 1-9 adalah langkah yang sama yang akan Anda gunakan untuk meluncurkan On-Demand Instance. Pada Langkah 10, Anda mengonfigurasi permintaan Instans Spot.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di bilah navigasi di bagian atas layar, pilih wilayah.
3. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.
4. (Opsional) Pada bagian Nama dan tanda, Anda dapat memberi nama pada instans, serta menandai permintaan instans Spot, instans, volume, dan grafik elastis. Untuk informasi tentang tanda, lihat Tandai sumber daya Amazon EC2 Anda.

   a. Untuk Nama, masukkan nama deskriptif untuk instans Anda.

      Nama instans adalah tanda, di mana kuncinya adalah Name, dan nilainya adalah nama yang Anda tentukan. Jika Anda tidak menentukan nama, instans dapat diidentifikasi berdasarkan ID-nya, yang secara otomatis dihasilkan saat Anda meluncurkan instans tersebut.

   b. Untuk menandai permintaan Instans Spot, instans, volume, dan grafik elastis, pilih Tambahkan tanda tambahan. Pilih Tambahkan tanda, lalu masukkan kunci dan nilai, lalu pilih jenis sumber daya yang akan diberi tanda. Pilih Tambah tanda lagi untuk setiap tanda tambahan yang akan ditambahkan.
5. Di bawah Citra Aplikasi dan OS (Amazon Machine Image), pilih sistem operasi (OS) untuk instans Anda, lalu pilih AMI. Untuk informasi selengkapnya, lihat Aplikasi dan Gambar OS (Gambar Mesin Amazon).
6. Di bawah tipe instans, pilih tipe instans yang memenuhi persyaratan Anda untuk konfigurasi perangkat keras dan ukuran instans Anda. Untuk informasi selengkapnya, lihat Tipe instans.
7. Di bawah Nama pasangan kunci (login), pilih pasangan kunci yang ada, atau pilih Buat pasangan kunci baru untuk membuat yang baru. Untuk informasi selengkapnya, lihat Pasangan kunci Amazon EC2 dan instans Amazon EC2.

> **Important**
>
> Jika Anda memilih opsi Lanjutkan tanpa pasangan kunci (Tidak direkomendasikan), Anda tidak akan dapat terhubung ke instans tersebut, kecuali Anda memilih sebuah AMI yang dikonfigurasi agar pengguna dapat masuk dengan cara lain.

8. Di bawah Pengaturan jaringan, gunakan pengaturan default, atau pilih Edit untuk mengonfigurasi pengaturan jaringan jika diperlukan.

   Grup keamanan membentuk bagian dari pengaturan jaringan dan menentukan aturan firewall untuk instans Anda. Aturan ini menentukan lalu lintas jaringan yang masuk yang dikirim ke instans Anda.

   Untuk informasi selengkapnya, lihat Pengaturan jaringan.

9. AMI yang Anda pilih mencakup satu atau lebih volume penyimpanan, termasuk volume perangkat root. Pada bagian Konfigurasikan penyimpanan, Anda dapat menentukan volume tambahan untuk dilampirkan ke instans dengan memilih Tambahkan volume baru. Untuk informasi selengkapnya, lihat Mengonfigurasi penyimpanan.

10. Pada bagian Detail lanjutan, konfigurasikan permintaan Instans Spot sebagai berikut:

    a. Di bawah opsi Pembel ian, pilih kotak centang Minta Instans Spot.

    b. Anda dapat menyimpan konfigurasi default untuk permintaan Instans Spot, atau memilih Sesuaikan (di sebelah kanan) agar dapat menentukan pengaturan khusus untuk permintaan Instans Spot Anda.

       Saat Anda memilih Sesuaikan, bidang berikut akan muncul.

       i. Harga maksimum: Anda dapat meminta Instans Spot dengan harga Spot, dibatasi pada On-Demand harga, atau Anda dapat menentukan jumlah maksimum yang bersedia Anda bayar.

          > **Warning**
          >
          > Jika Anda menentukan harga maksimum, instans Anda akan lebih sering diinterupsi daripada jika Anda memilih Tidak ada harga maksimum.

Jika Anda menentukan harga maksimum, itu harus lebih dari USD $0.001. Menentukan nilai di bawah USD $0.001 akan mengakibatkan peluncuran gagal.

- Tidak ada harga maksimum: Instans Spot Anda akan diluncurkan pada harga Spot saat ini. Harga tidak akan pernah melebihi On-Demand harga. (Direkomendasikan)
- Tetapkan harga maksimum Anda (per instance/hour): Anda dapat menentukan jumlah maksimum yang bersedia Anda bayar.
  - Jika Anda menentukan harga maksimum yang kurang dari harga Spot saat ini, Instans Spot Anda tidak akan diluncurkan.
  - Jika Anda menentukan harga maksimum melebihi harga Spot saat ini, Instans Spot Anda akan diluncurkan dan dikenai biaya sesuai harga Spot saat ini. Setelah Instans Spot berjalan, jika harga Spot naik di atas harga maksimum, Amazon EC2 akan menginterupsi Instans Spot Anda.
  - Berapa pun harga maksimum yang Anda tentukan, Anda akan selalu dikenai biaya sesuai harga Spot saat ini.

  Untuk meninjau tren harga Spot, lihat Lihat riwayat harga Instans Spot.

ii. Tipe permintaan: Permintaan Instans Spot yang dipilih menentukan apa yang terjadi jika Instans Spot Anda diinterupsi.

- One-time: Amazon EC2 mengajukan permintaan satu kali untuk Instans Spot Anda. Jika Instans Spot Anda diinterupsi, permintaan tidak akan dikirim ulang.
- Permintaan persisten: Amazon EC2 menempatkan permintaan persisten untuk Instans Spot Anda. Jika Instans Spot Anda diinterupsi, permintaan dikirimkan ulang untuk mengisi Instans Spot yang diinterupsi.

Jika Anda tidak menentukan nilai, default-nya adalah permintaan satu kali.

iii. Berlaku hingga: Tanggal kedaluwarsa dari permintaan Instans Spot persisten.

Bidang ini tidak didukung untuk permintaan satu kali. Permintaan satu kali tetap aktif hingga semua instans dalam permintaan diluncurkan atau Anda membatalkan permintaan.

- Tidak ada tanggal kedaluwarsa permintaan: Permintaan tetap aktif hingga Anda membatalkannya.
- Atur tanggal kedaluwarsa permintaan Anda: Permintaan persisten tetap aktif hingga tanggal yang Anda tentukan, atau sampai Anda membatalkannya.

iv. Perilaku interupsi: Perilaku yang Anda pilih menentukan apa yang terjadi saat Instans Spot diinterupsi.

- Untuk permintaan persisten, nilai yang valid adalah Berhenti dan Hibernasi. Saat instans dihentikan, biaya penyimpanan volume EBS diterapkan.

> **Note**
>
> Instans Spot sekarang menggunakan fungsionalitas hibernasi yang sama dengan On-Demand Instans. Untuk mengaktifkan hibernasi, Anda dapat memilih Hibernasi di sini, atau Anda dapat memilih Aktifkan dari bidang Perilaku Berhenti - Hibernasi, yang muncul lebih rendah di wizard peluncuran instans. Untuk prasyarat hibernasi, lihat Prasyarat untuk hibernasi instans EC2.

- Untuk permintaan satu kali, hanya Akhiri yang valid.

Jika Anda tidak menentukan nilai, default-nya Akhiri, yang tidak valid untuk permintaan Instans Spot yang persisten. Jika Anda mempertahankan default dan mencoba meluncurkan permintaan Instans Spot persisten, Anda akan mendapatkan pesan kesalahan.

Untuk informasi selengkapnya, lihat Perilaku interupsi Instance Spot.

11. Pada panel Ringkasan, untuk Jumlah instans, masukkan jumlah instans yang akan diluncurkan.

> **Note**
>
> Amazon EC2 membuat permintaan terpisah untuk setiap Instans Spot.

12. Pada panel Ringkasan, tinjau detail instans Anda, dan buat perubahan yang diperlukan. Setelah mengirimkan permintaan Instans Spot, Anda tidak dapat mengubah parameter

permintaan. Anda dapat secara langsung menavigasi ke bagian di wizard peluncuran instans dengan memilih tautannya di panel Ringkasan. Untuk informasi selengkapnya, lihat Ringkasan.

13. Ketika Anda siap untuk meluncurkan instans Anda, pilih Launch instans.

    Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

AWS CLI

Untuk membuat permintaan Instans Spot menggunakan run-instances

Gunakan perintah run-instance dan tentukan opsi Spot Instance dalam `--instance-market-options` parameter sebagai berikut.

```
--instance-market-options file://spot-options.json
```

Berikut ini adalah struktur data untuk menentukan dalam file JSON. Anda juga dapat menentukan `ValidUntil` dan `InstanceInterruptionBehavior`. Jika Anda tidak menentukan bidang dalam struktur data, maka nilai default yang akan digunakan.

Contoh berikut membuat permintaan `persistent`.

```
{
  "MarketType": "spot",
  "SpotOptions": {
    "SpotInstanceType": "persistent"
  }
}
```

Untuk membuat permintaan Instance Spot menggunakan request-spot-instance

> **Note**
>
> Kami sangat tidak menyarankan menggunakan perintah request-spot-instanss untuk meminta instans Spot karena ini adalah API lama tanpa investasi yang direncanakan. Untuk informasi selengkapnya, lihat Metode permintaan Spot mana yang terbaik untuk digunakan?

Gunakan perintah request-spot-instances untuk membuat permintaan satu kali.

```
aws ec2 request-spot-instances \
    --instance-count 5 \
    --type "one-time" \
    --launch-specification file://specification.json
```

Gunakan perintah request-spot-instances untuk membuat permintaan persisten.

```
aws ec2 request-spot-instances \
    --instance-count 5 \
    --type "persistent" \
    --launch-specification file://specification.json
```

Misalnya, file spesifikasi peluncuran untuk digunakan dengan perintah ini, lihat Contoh spesifikasi peluncuran permintaan Instans Spot. Jika Anda mengunduh file spesifikasi peluncuran dari konsol Permintaan Spot, Anda harus menggunakan perintah request-spot-fleet (konsol Permintaan Spot menentukan permintaan Instans Spot menggunakan Armada Spot).

PowerShell

Untuk membuat permintaan Instance Spot

Gunakan New-EC2Instance cmdlet dan tentukan opsi Spot Instance menggunakan parameter. `-InstanceMarketOption`

```
-InstanceMarketOptions $marketOptions
```

Buat struktur data untuk opsi Instance Spot sebagai berikut.

```
$spotOptions = New-Object Amazon.EC2.Model.SpotMarketOptions
$spotOptions.SpotInstanceType="persistent"
$marketOptions = New-Object Amazon.EC2.Model.InstanceMarketOptionsRequest
$marketOptions.MarketType = "spot"
$marketOptions.SpotOptions = $spotOptions
```

### Contoh spesifikasi peluncuran permintaan Instans Spot

Contoh berikut menunjukkan konfigurasi peluncuran yang dapat Anda gunakan dengan perintah request-spot-instances untuk membuat permintaan Instans Spot. Untuk informasi selengkapnya, lihat Mengelola Instans Spot Anda.

> **Important**
>
> Kami sangat tidak menyarankan menggunakan perintah request-spot-instanss untuk meminta instans Spot karena ini adalah API lama tanpa investasi yang direncanakan. Untuk informasi selengkapnya, lihat Metode permintaan Spot mana yang terbaik untuk digunakan?

Contoh

- Contoh 1: Luncurkan Instans Spot
- Contoh 2: Luncurkan Instans Spot dalam Zona Ketersediaan yang ditentukan
- Contoh 3: Luncurkan Instans Spot di subnet yang ditentukan
- Contoh 4: Luncurkan Instans Spot Khusus

Contoh 1: Luncurkan Instans Spot

Contoh berikut tidak menyertakan Zona Ketersediaan atau subnet. Amazon EC2 memilih Zona Ketersediaan untuk Anda. Amazon EC2 meluncurkan instans di subnet default dari Zona Ketersediaan yang dipilih.

```
{
  "ImageId": "ami-0abcdef1234567890",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
  "InstanceType": "m5.medium",
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Contoh 2: Luncurkan Instans Spot dalam Zona Ketersediaan yang ditentukan

Contoh berikut mencakup Zona Ketersediaan. Amazon EC2 meluncurkan instans di subnet default dari Zona Ketersediaan yang dipilih.

```
{
  "ImageId": "ami-0abcdef1234567890",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
```

```
  "InstanceType": "m5.medium",
  "Placement": {
    "AvailabilityZone": "us-west-2a"
  },
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Contoh 3: Luncurkan Instans Spot di subnet yang ditentukan

Contoh berikut menyertakan subnet. Amazon EC2 meluncurkan instans di subnet yang ditentukan. Jika VPC adalah VPC nondefault, instans tidak akan menerima alamat IPv4 publik secara default.

```
{
  "ImageId": "ami-0abcdef1234567890",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
  "InstanceType": "m5.medium",
  "SubnetId": "subnet-1a2b3c4d",
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Untuk menetapkan alamat IPv4 publik ke sebuah instans dalam VPC nondefault, tentukan bidang `AssociatePublicIpAddress`seperti yang ditunjukkan pada contoh berikut. Saat Anda menentukan antarmuka jaringan, Anda harus menyertakan ID subnet dan ID grup keamanan menggunakan antarmuka jaringan, daripada menggunakan bidang `SubnetId` dan `SecurityGroupIds` seperti yang ditunjukkan dalam blok kode sebelumnya.

```
{
  "ImageId": "ami-0abcdef1234567890",
  "KeyName": "my-key-pair",
  "InstanceType": "m5.medium",
  "NetworkInterfaces": [
    {
      "DeviceIndex": 0,
      "SubnetId": "subnet-1a2b3c4d5e6f7g8h9",
      "Groups": [ "sg-1a2b3c4d5e6f7g8h9" ],
      "AssociatePublicIpAddress": true
    }
  ],
```

```
  "IamInstanceProfile": {
      "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
  }
}
```

Contoh 4: Luncurkan Instans Spot Khusus

Contoh berikut meminta Instans Spot dengan penghunian `dedicated`. Instans Spot Khusus harus diluncurkan di VPC.

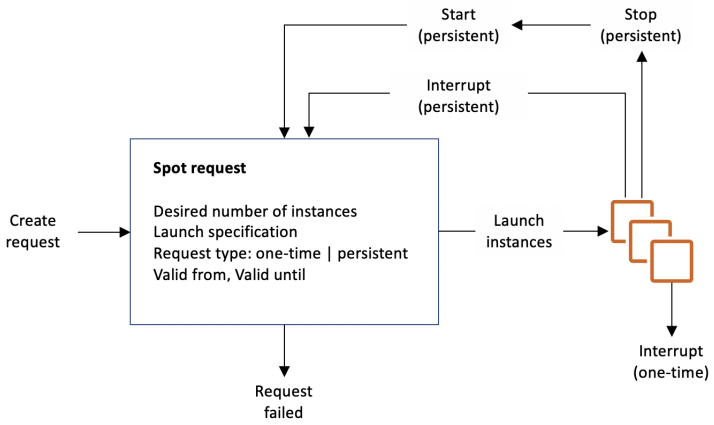
```
{
  "ImageId": "ami-0abcdef1234567890",
  "KeyName": "my-key-pair",
  "SecurityGroupIds": [ "sg-1a2b3c4d5e6f7g8h9" ],
  "InstanceType": "c5.8xlarge",
  "SubnetId": "subnet-1a2b3c4d5e6f7g8h9",
  "Placement": {
    "Tenancy": "dedicated"
  }
}
```

## Mendapatkan status permintaan Instance Spot

Untuk membantu Anda melacak permintaan Instans Spot dan merencanakan penggunaan Instans Spot, gunakan status permintaan yang disediakan oleh Amazon EC2. Misalnya, status permintaan dapat memberikan alasan mengapa permintaan Spot Anda belum terpenuhi, atau mencantumkan kendala yang mencegah pemenuhan permintaan Spot Anda.

Pada setiap langkah proses—disebut juga dengan siklus hidup permintaan Spot—peristiwa spesifik menentukan status permintaan secara berurutan.

Ilustrasi berikut menunjukkan cara kerja permintaan Instans Spot. Perhatikan bahwa tipe permintaan (satu kali atau tetap) menentukan apakah permintaan dibuka lagi saat Amazon EC2 menginterupsi Instans Spot atau jika Anda menghentikan Instans Spot. Jika permintaan tetap ada, permintaan dibuka lagi setelah Instans Spot Anda diinterupsi. Jika permintaan tetap ada dan Anda menghentikan Instans Spot, permintaan tersebut hanya terbuka setelah Anda memulai Instans Spot.




Daftar Isi


- Dapatkan informasi status permintaan
- Kode status permintaan Spot
- Peristiwa Pemenuhan Permintaan Instans Spot EC2
- Perubahan status untuk permintaan Spot


## Dapatkan informasi status permintaan

Anda bisa mendapatkan informasi status untuk permintaan Instans Spot Anda.

### Console

Untuk mendapatkan informasi status permintaan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot, lalu pilih permintaan Spot.
3. Untuk memeriksa status, pada tab Deskripsi, periksa bidang Status.

### AWS CLI

Untuk mendapatkan informasi status permintaan

Gunakan perintah describe-spot-instance-quests berikut.

```
aws ec2 describe-spot-instance-requests --spot-instance-request-
ids sir-0e54a519c9EXAMPLE
```

PowerShell

Untuk mendapatkan informasi status permintaan

Gunakan Get-EC2SpotInstanceRequest cmdlet.

```
Get-EC2SpotInstanceRequest -SpotInstanceRequestId sir-0e54a519c9EXAMPLE
```

Kode status permintaan Spot

Informasi status permintaan Spot terdiri dari kode status, waktu pembaruan, dan pesan status. Semua itu membantu Anda menentukan disposisi permintaan Spot Anda.

Berikut ini adalah kode status permintaan Spot:

`az-group-constraint`

Amazon EC2 tidak dapat meluncurkan semua instans yang Anda minta di Zona Ketersediaan yang sama.

`bad-parameters`

Satu atau lebih parameter untuk permintaan Spot Anda tidak valid (misalnya, AMI yang Anda tentukan tidak ada). Pesan status menunjukkan parameter mana yang tidak valid.

`canceled-before-fulfillment`

Pengguna membatalkan permintaan Spot sebelum permintaan dipenuhi.

`capacity-not-available`

Tidak tersedia kapasitas yang cukup untuk instans yang Anda minta.

`constraint-not-fulfillable`

Permintaan Spot tidak dapat dipenuhi karena satu atau beberapa batasan tidak valid (misalnya, Zona Ketersediaan tidak ada). Pesan status menunjukkan batasan mana yang tidak valid.

`fulfilled`

Permintaan Spot adalah `active`, dan Amazon EC2 meluncurkan Instans Spot Anda.

`instance-stopped-by-price`

Instans Anda berhenti karena harga Spot melebihi harga maksimum Anda.

`instance-stopped-by-user`

Instans Anda berhenti karena pengguna menghentikan instans atau menjalankan perintah penonaktifan dari instans tersebut.

`instance-stopped-no-capacity`

Instans Anda berhenti karena kebutuhan manajemen kapasitas EC2.

`instance-terminated-by-price`

Instans Anda diakhiri karena harga Spot melebihi harga maksimum Anda. Jika permintaan Anda persisten, prosesnya akan dimulai ulang, jadi permintaan Anda menunggu evaluasi.

`instance-terminated-by-schedule`

Instans Spot Anda diakhiri di akhir durasi yang dijadwalkan.

`instance-terminated-by-service`

Instans Anda dihentikan dari status berhenti.

`instance-terminated-by-user` atau `spot-instance-terminated-by-user`

Anda mengakhiri Instans Spot yang telah terpenuhi, jadi status permintaannya adalah `closed` (kecuali permintaan persisten) dan status instans adalah `terminated`.

`instance-terminated-launch-group-constraint`

Satu atau beberapa instans dalam grup peluncuran Anda telah diakhiri, sehingga batasan grup peluncuran tidak lagi dipenuhi.

`instance-terminated-no-capacity`

Instans Anda diakhiri karena proses manajemen kapasitas standar.

`launch-group-constraint`

Amazon EC2 tidak dapat meluncurkan semua instans yang Anda minta pada saat yang bersamaan. Semua instans dalam grup peluncuran dimulai dan diakhiri bersama.

`limit-exceeded`

Batas jumlah volume EBS atau total volume penyimpanan telah terlampaui. Untuk informasi selengkapnya, lihat Kuota untuk Amazon EBS di Panduan Pengguna Amazon EBS.

`marked-for-stop`

Instans Spot ditandai karena berhenti.

`marked-for-termination`

Instans Spot ditandai karena pengakhiran.

`not-scheduled-yet`

Permintaan Spot tidak dievaluasi hingga tanggal yang dijadwalkan.

`pending-evaluation`

Setelah Anda membuat permintaan Instans Spot, permintaan itu masuk dalam status `pending-evaluation` sementara sistem mengevaluasi parameter permintaan Anda.

`pending-fulfillment`

Amazon EC2 mencoba menyediakan Instans Spot Anda.

`placement-group-constraint`

Permintaan Spot belum dapat dipenuhi karena Instans Spot tidak dapat ditambahkan ke grup penempatan saat ini.

`price-too-low`

Permintaan belum dapat dipenuhi karena harga maksimum Anda di bawah harga Spot. Dalam kasus ini, tidak ada instans yang diluncurkan dan permintaan Anda tetap `open`.

`request-canceled-and-instance-running`

Anda membatalkan permintaan Spot saat Instans Spot masih berjalan. Permintaannya `cancelled`, tapi instans tetap `running`.

`schedule-expired`

Permintaan Spot kedaluwarsa karena tidak terpenuhi sebelum tanggal yang ditentukan.

`system-error`

Terjadi kesalahan sistem yang tidak terduga. Jika ini adalah masalah berulang, silakan hubungi AWS Dukungan untuk bantuan.

### Peristiwa Pemenuhan Permintaan Instans Spot EC2

Saat permintaan Instans Spot dipenuhi, Amazon EC2 mengirimkan acara Pemenuhan Permintaan Instans Spot EC2 ke Amazon. EventBridge Anda dapat membuat aturan untuk mengambil tindakan

kapan pun peristiwa ini terjadi, seperti menginvokasi fungsi Lambda atau memberi tahu topik Amazon SNS.

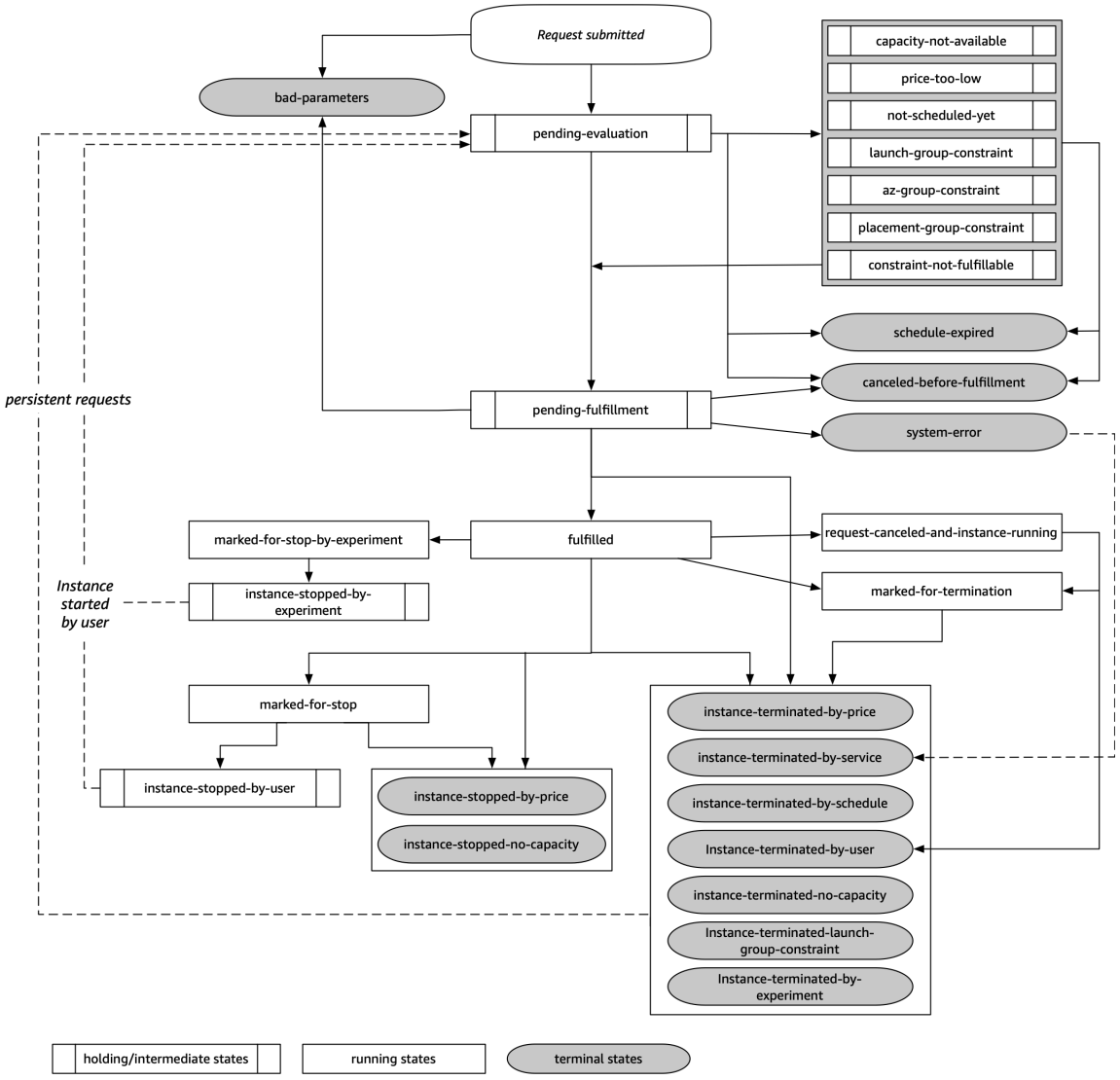
Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "01234567-1234-0123-1234-012345678901",
    "detail-type": "EC2 Spot Instance Request Fulfillment",
    "source": "aws.ec2",
    "account": "123456789012",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890abcdef0"],
    "detail": {
        "spot-instance-request-id": "sir-0e54a519c9EXAMPLE",
        "instance-id": "i-1234567890abcdef0"
    }
}
```

Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Perubahan status untuk permintaan Spot

Diagram berikut menunjukkan kepada Anda jalur yang dapat diikuti oleh permintaan Spot Anda sepanjang siklus hidupnya, dari pengiriman hingga pengakhiran. Setiap langkah digambarkan sebagai suatu simpul, dan kode status untuk setiap simpul menjelaskan status permintaan Spot dan Instans Spot.




## Evaluasi tertunda

Segera setelah Anda membuat permintaan Instans Spot, permintaan itu masuk ke status `pending-evaluation` kecuali jika ada satu atau lebih parameter permintaan yang tidak valid (`bad-parameters`).

| Kode status | Status permintaan | Status instans |
|---|---|---|
| `pending-evaluation` | open | Tidak berlaku |
| `bad-parameters` | `closed` | Tidak berlaku |

## Menunggu

Jika satu atau beberapa batasan permintaan sudah valid tetapi belum dapat dipenuhi, atau jika kapasitas tidak mencukupi, permintaan masuk ke status menunggu sampai batasan tersebut terpenuhi. Opsi permintaan memengaruhi kemungkinan permintaan dipenuhi. Misalnya, jika tidak ada kapasitas, permintaan Anda akan tetap dalam status menunggu hingga ada kapasitas yang tersedia. Jika Anda menentukan grup Zona Ketersediaan, permintaan tetap dalam status menunggu hingga batasan Zona Ketersediaan terpenuhi.

Jika terjadi pemadaman di salah satu Zona Ketersediaan, ada kemungkinan kapasitas EC2 cadangan yang tersedia untuk permintaan Instans Spot di Zona Ketersediaan lainnya dapat terpengaruh.

| Kode status | Status permintaan | Status instans |
|---|---|---|
| `capacity-not-available` | `open` | Tidak berlaku |
| `price-too-low` | `open` | Tidak berlaku |
| `not-scheduled-yet` | `open` | Tidak berlaku |
| `launch-group-constraint` | `open` | Tidak berlaku |
| `az-group-constraint` | `open` | Tidak berlaku |
| `placement-group-constraint` | `open` | Tidak berlaku |
| `constraint-not-fulfillable` | `open` | Tidak berlaku |

Tertunda evaluation/fulfillment -terminal

Permintaan Instans Spot Anda dapat masuk ke status `terminal` jika Anda membuat permintaan yang valid hanya selama jangka waktu tertentu dan jangka waktu ini berakhir sebelum permintaan Anda mencapai fase pemenuhan tertunda. Mungkin juga terjadi jika Anda membatalkan permintaan, atau jika terjadi kesalahan sistem.

| Kode status | Status permintaan | Status instans |
| --- | --- | --- |
| `schedule-expired` | `cancelled` | Tidak berlaku |
| `canceled-before-fulfillment` [1] | `cancelled` | Tidak berlaku |
| `bad-parameters` | `failed` | Tidak berlaku |
| `system-error` | `closed` | Tidak berlaku |

[1] Jika Anda membatalkan permintaan.

Pemenuhan tertunda

Ketika batasan yang Anda tentukan (jika ada) terpenuhi, permintaan Spot Anda masuk ke status `pending-fulfillment`.

Pada titik ini, Amazon EC2 sedang bersiap untuk menyediakan instans yang Anda minta. Jika proses berhenti pada titik ini, kemungkinan besar karena proses itu dibatalkan oleh pengguna sebelum Instans Spot diluncurkan. Hal ini mungkin juga karena terjadi kesalahan sistem yang tidak terduga.

| Kode status | Status permintaan | Status instans |
| --- | --- | --- |
| `pending-fulfillment` | `open` | Tidak berlaku |

Terpenuhi

Saat semua spesifikasi untuk Instans Spot Anda terpenuhi, permintaan Spot Anda dipenuhi. Amazon EC2 meluncurkan Instans Spot, yang dapat memerlukan waktu beberapa menit. Jika Instans Spot menjalani hibernasi atau berhenti saat diinterupsi, Instans Spot tetap dalam status ini hingga permintaan dapat dipenuhi lagi atau permintaan dibatalkan.

| Kode status | Status permintaan | Status instans |
| --- | --- | --- |
| `fulfilled` | `active` | `pending` → `running` |
| `fulfilled` | `active` | `stopped` → `running` |

Jika Anda menghentikan Instans Spot, permintaan Spot Anda akan masuk dalam status `marked-for-stop` atau `instance-stopped-by-user` hingga Instans Spot dapat dimulai lagi atau permintaan dibatalkan.

| Kode status | Status permintaan | Status instans |
| --- | --- | --- |
| `marked-for-stop` | `active` | `stopping` |
| `instance-stopped-by-user`[1] | `disabled` atau `cancelled`[2] | `stopped` |

* Instans Spot masuk dalam status `instance-stopped-by-user` jika Anda menghentikan instans atau menjalankan perintah pematian dari instans. Setelah Anda menghentikan instans, Anda dapat memulainya lagi. Saat memulai ulang, permintaan Instans Spot kembali ke status `pending-evaluation` dan Amazon EC2 meluncurkan Instans Spot baru ketika batasan terpenuhi.

[2] Status permintaan Spot adalah `disabled` jika Anda menghentikan Instans Spot tetapi tidak membatalkan permintaan. Status permintaan adalah `cancelled` jika Instans Spot Anda dihentikan dan permintaan kedaluwarsa.

Fulfilled-terminal

Instans Spot Anda terus berjalan selama ada kapasitas yang tersedia untuk tipe instans Anda, dan Anda tidak mengakhiri instans. Jika Amazon EC2 harus mengakhiri Instans Spot Anda, permintaan Spot masuk ke status terminal. Permintaan juga masuk ke status terminal jika Anda membatalkan permintaan Spot atau mengakhiri Instans Spot.

| Kode status | Status permintaan | Status instans |
| --- | --- | --- |
| `request-canceled-and-instance-running` | `cancelled` | `running` |

| Kode status | Status permintaan | Status instans |
| --- | --- | --- |
| marked-for-stop | active | running |
| marked-for-termination | active | running |
| instance-stopped-by-price | disabled | stopped |
| instance-stopped-by-user | disabled | stopped |
| instance-stopped-no-capacity | disabled | stopped |
| instance-terminated-by-price | closed(satu kali),open (gigih) | terminated |
| instance-terminated-by-schedule | closed | terminated |
| instance-terminated-by-service | cancelled | terminated |
| instance-terminated-by-user | closed atau cancelled [1] | terminated |
| instance-terminated-no-capacity | closed(satu kali),open (gigih) | running † |
| instance-terminated-no-capacity | closed(satu kali),open (gigih) | terminated |
| instance-terminated-launch-group-constraint | closed(satu kali),open (gigih) | terminated |

* Status permintaan adalah `closed` jika Anda mengakhiri instans, tetapi tidak membatalkan permintaan. Status permintaan adalah `cancelled` jika Anda mengakhiri instans dan membatalkan permintaan. Meskipun Anda mengakhiri Instans Spot sebelum Anda membatalkan permintaannya, penundaan mungkin terjadi sebelum Amazon EC2 mendeteksi bahwa Instans Spot Anda telah diakhiri. Dalam hal ini, status permintaan bisa berupa `closed` atau `cancelled`.

† Saat Amazon EC2 menginterupsi Instans Spot karena memerlukan kapasitas kembali dan instans dikonfigurasi untuk berakhir saat terjadi interupsi, status akan segera diatur ke `instance-terminated-no-capacity` (tidak diatur ke `marked-for-termination`). Namun, instans tetap dalam status `running` selama 2 menit untuk mencerminkan periode 2 menit saat instans menerima pemberitahuan interupsi Instans Spot. Setelah 2 menit, status instans diatur ke `terminated`.

Eksperimen interupsi

Anda dapat menggunakannya AWS Fault Injection Service untuk memulai interupsi Instans Spot sehingga Anda dapat menguji bagaimana aplikasi di Instans Spot Anda merespons. Jika AWS FIS menghentikan Instans Spot, permintaan Spot Anda memasuki `marked-for-stop-by-experiment` status dan kemudian `instance-stopped-by-experiment` status. Jika AWS FIS mengakhiri Instans Spot, permintaan Spot Anda memasuki `instance-terminated-by-experiment` status. Untuk informasi selengkapnya, lihat the section called “Memulai interupsi”.

| Kode status | Status permintaan | Status instance |
|---|---|---|
| `marked-for-stop-by-experiment` | `active` | `running` |
| `instance-stopped-by-experiment` | `disabled` | `stopped` |
| `instance-terminated-by-experiment` | `closed` | `terminated` |

Permintaan yang persisten

Saat Instans Spot Anda diakhiri (baik oleh Anda maupun Amazon EC2), jika permintaan Spot adalah permintaan yang persisten, instans akan kembali ke status `pending-evaluation` dan Amazon EC2 dapat meluncurkan Instans Spot baru saat batasan terpenuhi.

## Menandai permintaan Instans Spot

Untuk membantu mengategorikan dan mengelola permintaan Instans Spot, Anda dapat menandainya dengan metadata kustom. Anda dapat menetapkan tanda untuk permintaan Instans Spot saat Anda membuatnya, atau setelahnya. Anda dapat menetapkan tanda menggunakan konsol Amazon EC2 atau alat baris perintah.

Saat Anda menandai permintaan Instans Spot, instans dan volume yang diluncurkan oleh Instans Spot tidak secara otomatis ditandai. Anda perlu menandai instans dan volume yang diluncurkan oleh Instans Spot secara eksplisit. Anda dapat menetapkan tanda ke Instans Spot dan volume selama peluncuran, atau setelahnya.

Untuk informasi selengkapnya tentang cara kerja tag, lihat Tandai sumber daya Amazon EC2 Anda.

Daftar Isi


- Prasyarat
- Menandai permintaan Instans Spot baru
- Menandai permintaan Instans Spot yang ada
- Melihat tanda permintaan Instans Spot


### Prasyarat

Berikan izin kepada pengguna untuk menandai sumber daya. Untuk informasi selengkapnya tentang kebijakan IAM dan contoh kebijakan, lihat Contoh: Memberi tanda pada sumber daya.

Kebijakan IAM yang Anda buat ditentukan oleh metode yang Anda gunakan untuk membuat permintaan Instans Spot.

- Jika Anda menggunakan wizard peluncuran instans atau `run-instances` untuk meminta Instans Spot, lihat To grant a user the permission to tag resources when using the launch instance wizard or run-instances.
- Jika Anda menggunakan perintah `request-spot-instances` untuk meminta Instans Spot, lihat To grant a user the permission to tag resources when using request-spot-instances.

Untuk memberikan izin menandai sumber daya kepada pengguna saat menggunakan wizard peluncuran instans atau run-instances

Buat kebijakan IAM yang mencakup hal-hal berikut:

- Tindakan `ec2:RunInstances`. Tindaka ini memberikan izin kepada pengguna untuk meluncurkan sebuah instans.
- Untuk, `Resource`tentukan.`spot-instances-request` Ini memungkinkan pengguna untuk membuat permintaan Instans Spot, yang meminta Instans Spot.
- Tindakan `ec2:CreateTags`. Tindakan ini memberikan izin kepada pengguna untuk membuat tanda.
- Untuk `Resource`, tentukan `*`. Hal ini memungkinkan para pengguna untuk menandai semua sumber daya yang dibuat selama peluncuran.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowLaunchInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "TagSpotInstanceRequests",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

Saat Anda menggunakan RunInstances tindakan untuk membuat permintaan Instans Spot dan menandai permintaan Instans Spot saat dibuat, Anda harus mengetahui cara Amazon EC2 mengevaluasi `spot-instances-request` sumber daya dalam RunInstances pernyataan yang dievaluasi dalam kebijakan IAM sebagai berikut:

- Jika Anda tidak memberi tag pada permintaan Instans Spot saat dibuat, Amazon EC2 tidak mengevaluasi`spot-instances-request` sumber daya di RunInstances pernyataan.
- Jika Anda memberi tag pada permintaan Instans Spot saat dibuat, Amazon EC2 mengevaluasi`spot-instances-request` sumber daya di RunInstances pernyataan.

Oleh karena itu, untuk sumber daya `spot-instances-request`, aturan-aturan berikut berlaku untuk kebijakan IAM:

- Jika Anda menggunakan RunInstances untuk membuat permintaan Instance Spot dan Anda tidak bermaksud untuk menandai permintaan Instans Spot saat dibuat, Anda tidak perlu mengizinkan `spot-instances-request` sumber daya secara eksplisit; panggilan akan berhasil.
- Jika Anda menggunakan RunInstances untuk membuat permintaan Instance Spot dan ingin menandai permintaan Instans Spot saat dibuat, Anda harus menyertakan `spot-instances-request` sumber daya dalam pernyataan RunInstances allow, jika tidak panggilan akan gagal.
- Jika Anda menggunakan RunInstances untuk membuat permintaan Instans Spot dan ingin menandai permintaan Instans Spot saat dibuat, Anda harus menentukan `spot-instances-request` sumber daya atau menyertakan `*` wildcard dalam pernyataan CreateTags allow, jika tidak panggilan akan gagal.

Misalnya kebijakan IAM, termasuk kebijakan yang tidak didukung untuk permintaan Instans Spot, lihat Cara Menggunakan Instans Spot.

Untuk memberikan izin menandai sumber daya kepada pengguna saat menggunakan request-spot-instances

Buat kebijakan IAM yang mencakup hal-hal berikut:

- Tindakan `ec2:RequestSpotInstances`. Tindakan ini memberikan izin kepada pengguna untuk membuat permintaan Instans Spot.
- Tindakan `ec2:CreateTags`. Tindakan ini memberikan izin kepada pengguna untuk membuat tanda.

- Untuk `Resource`, tentukan `spot-instances-request`. Hal ini memungkinkan pengguna untuk hanya menandai permintaan Instans Spot.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "TagSpotInstanceRequest",
            "Effect": "Allow",
            "Action": [
                "ec2:RequestSpotInstances",
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-instances-
request/*"
        }
    ]
}
```

Menandai permintaan Instans Spot baru

Dalam PowerShell contoh AWS CLI dan, konfigurasikan permintaan Instance Spot sebagai berikut:

- Untuk `ResourceType`, tentukan `spot-instances-request`. Jika Anda menentukan nilai lain, maka permintaan Instans Spot akan gagal.
- Untuk `Tags`, tentukan pasangan nilai-kunci. Anda dapat menentukan lebih dari satu pasangan nilai-kunci.

Console

Untuk menandai permintaan Instance Spot baru

1. Ikuti prosedur Mengelola Instans Spot Anda.
2. Untuk menambahkan tanda, pada halaman Tambahkan Tanda, pilih Tambahkan Tanda, lalu masukkan kunci dan nilai untuk tanda tersebut. Pilih Tambahkan tanda lain untuk setiap tanda tambahan.

Untuk setiap tanda, Anda dapat menandai permintaan Instans Spot, Instans Spot, dan volume dengan tanda yang sama. Untuk menandai ketiganya, pastikan bahwa Instans, Volume, dan Permintaan Instans Spot telah dipilih. Untuk menandai hanya satu atau dua, pastikan bahwa sumber daya yang ingin Anda tandai telah dipilih, dan pilihan pada sumber daya lainnya dihapus.

3. Lengkapi bidang yang diperlukan untuk membuat permintaan Instans Spot, lalu pilih Luncurkan. Untuk informasi selengkapnya, lihat Mengelola Instans Spot Anda.

AWS CLI

Untuk menandai permintaan Instance Spot baru

Gunakan perintah request-spot-instance dengan opsi. `--tag-specification`

Spesifikasi tag menambahkan dua tag ke permintaan Instance Spot: `Environment=Production` dan`Cost-Center=123`.

```
aws ec2 request-spot-instances \
    --instance-count 5 \
    --type "one-time" \
    --launch-specification file://specification.json \
    --tag-specification 'ResourceType=spot-instances-
request,Tags=[{Key=Environment,Value=Production},{Key=Cost-Center,Value=123}]'
```

PowerShell

Untuk menandai permintaan Instance Spot baru

Gunakan Request-EC2SpotInstance cmdlet dengan parameter. `-TagSpecification`

```
-TagSpecification $tagspec
```

Spesifikasi tag didefinisikan sebagai berikut. Ini menambahkan dua tag ke permintaan Instance Spot: `Environment=Production` dan`Cost-Center=123`.

```
$tag1 = @{Key="Environment"; Value="Production"}
$tag2 = @{Key="Cost-Center"; Value="123"}
$tagspec = New-Object Amazon.EC2.Model.TagSpecification
$tagspec.ResourceType = "spot-instances-request"
$tagspec.Tags = @($tag1,$tag2)
```

### Menandai permintaan Instans Spot yang ada

#### Console

Untuk menandai permintaan Instance Spot yang ada

Setelah Anda membuat permintaan Instans Spot, Anda dapat menambahkan tanda ke permintaan Instans Spot menggunakan konsol.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Instans Spot Anda.
4. Pilih tab Tanda dan pilih Buat Tanda.

Untuk menandai permintaan Instans Spot yang sudah ada menggunakan konsol

Setelah permintaan Instans Spot Anda meluncurkan Instans Spot, Anda dapat menambahkan tanda ke instans menggunakan konsol. Untuk informasi selengkapnya, lihat Tambahkan tag menggunakan konsol.

#### AWS CLI

Untuk menandai permintaan Instance Spot atau Instance Spot yang ada

Gunakan perintah create-tags untuk menandai sumber daya yang ada. Dalam contoh berikut, permintaan Instans Spot yang ada dan Instans Spot diberi tag dengan`purpose=test`.

```
aws ec2 create-tags \
    --resources sir-0e54a519c9EXAMPLE i-1234567890abcdef0 \
    --tags Key=purpose,Value=test
```

#### PowerShell

Untuk menandai permintaan Instance Spot atau Instance Spot yang ada

Gunakan New-EC2Tag cmdlet. Contoh berikut menambahkan tag `purpose=test` ke permintaan Instance Spot yang ada dan Instans Spot.

```
New-EC2Tag `
```

```
    -Resource sir-0e54a519c9EXAMPLE, i-1234567890abcdef0 `
    -Tag @{Key="purpose"; Value="test"}
```

### Melihat tanda permintaan Instans Spot

#### Console

Untuk melihat tag permintaan Instance Spot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Instans Spot Anda dan pilih tab Tanda.

#### AWS CLI

Untuk mendeskripsikan tag permintaan Instans Spot

Gunakan perintah describe-spot-instance-requests untuk melihat konfigurasi permintaan Instans Spot yang ditentukan, yang menyertakan tanda apa pun yang ditentukan untuk permintaan tersebut.

```
aws ec2 describe-spot-instance-requests \
    --spot-instance-request-ids sir-0e54a519c9EXAMPLE \
    --query "SpotInstanceRequests[*].Tags"
```

Berikut ini adalah output contoh.

```
[
    [
        {
            "Key": "Environment",
            "Value": "Production"
        },
        {
            "Key": "Department",
            "Value": "101"
        }
    ]
]
```

PowerShell

Untuk mendeskripsikan tag permintaan Instans Spot

Gunakan Get-EC2SpotInstanceRequest cmdlet.

```
(Get-EC2SpotInstanceRequest `
    -SpotInstanceRequestId sir-0e54a519c9EXAMPLE).Tags
```

Berikut ini adalah output contoh.

```
Key         Value
---         -----
Environment Production
Department  101
```

## Membatalkan permintaan Instans Spot

Jika Anda tidak lagi menginginkan permintaan Instans Spot, Anda dapat membatalkannya. Anda hanya dapat membatalkan permintaan Instans Spot yang `open`, `active`, atau `disabled`.

- Permintaan Instans Spot Anda adalah `open` saat permintaan Anda belum dipenuhi dan belum ada instans yang diluncurkan.
- Permintaan Instans Spot Anda adalah `active` saat permintaan Anda telah dipenuhi sehingga Instans Spot telah diluncurkan.
- Permintaan Instans Spot Anda adalah `disabled` saat Anda menghentikan Instans Spot Anda.

Jika permintaan Instans Spot Anda adalah `active` dan memiliki Instans Spot terkait yang sedang berjalan, membatalkan permintaan tidak akan menghentikan instans tersebut. Untuk informasi selengkapnya tentang pengakhiran Instans Spot, lihat Menghentikan Instans Spot.

Console

Untuk membatalkan permintaan Instance Spot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Instance Spot.

4. Pilih Tindakan, Batalkan permintaan.
5. (Opsional) Jika Anda telah selesai menggunakan Instans Spot terkait, Anda dapat mengakhirinya. Dalam kotak dialog Batalkan permintaan Spot, pilih Akhiri instans, lalu pilih Konfirmasi.

AWS CLI

Untuk membatalkan permintaan Instance Spot

Gunakan perintah cancel-spot-instance-quests berikut.

```
aws ec2 cancel-spot-instance-requests --spot-instance-request-
ids sir-0e54a519c9EXAMPLE
```

PowerShell

Untuk membatalkan permintaan Instance Spot

Gunakan Stop-EC2SpotInstanceRequest cmdlet.

```
Stop-EC2SpotInstanceRequest -SpotInstanceRequestId sir-0e54a519c9EXAMPLE
```

## Mengelola Instans Spot Anda

Amazon EC2 meluncurkan Instans Spot saat kapasitas tersedia. Instans Spot berjalan hingga diinterupsi atau Anda mengakhirinya sendiri.

Daftar Isi


- Menemukan Instans Spot Anda
- Menemukan instans yang diluncurkan oleh permintaan tertentu
- Menghentikan Instans Spot
- Memulai Instans Spot
- Menghentikan Instans Spot


### Menemukan Instans Spot Anda

Instans Spot muncul di halaman Instans di konsol, bersama dengan On-Demand Instans. Gunakan prosedur berikut untuk menemukan Instans Spot Anda.

Console

Untuk menemukan Instans Spot Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Untuk menemukan semua Instans Spot, di panel pencarian, pilih Instance lifecy cle=spot.
4. Untuk memverifikasi bahwa instance adalah Instans Spot, pilih instans, pilih tab Detail, dan periksa nilai Sik lus Hidup. Nilai untuk Instance Spot adalah `spot` dan nilai untuk On-Demand Instance adalah`normal`.

AWS CLI

Untuk menemukan Instans Spot Anda

Gunakan perintah describe-instance berikut.

```
aws ec2 describe-instances --filters "Name=instance-lifecycle,Values=spot"
```

Untuk menentukan apakah sebuah instance adalah Instance Spot

Gunakan perintah describe-instance berikut.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query "Reservations[*].Instances[*].InstanceLifecycle" \
    --output text
```

Jika outputnya adalah`spot`, instance tersebut adalah Instance Spot. Jika tidak ada output, instance tersebut adalah On-Demand Instance.

PowerShell

Untuk menemukan Instans Spot

Gunakan Get-EC2Instance cmdlet.

```
Get-EC2Instance -Filter @{Name="instance-lifecycle"; Values="spot"}
```

Untuk menentukan apakah sebuah instance adalah Instance Spot

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances.InstanceLifecycle
```

Jika outputnya adalah`Spot`, instance tersebut adalah Instance Spot. Jika tidak ada output, instance tersebut adalah On-Demand Instance.

Menemukan instans yang diluncurkan oleh permintaan tertentu

Gunakan prosedur berikut untuk menemukan Instans Spot yang diluncurkan dari Instans Spot atau permintaan Spot Fleet tertentu.

Console

Untuk menemukan Instans Spot untuk permintaan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot. Daftar ini berisi permintaan Instans Spot dan permintaan Spot Fleet.
3. Jika permintaan Instans Spot dipenuhi, Kapasitas adalah ID Instans Spot. Untuk Armada Spot, Kapasitas menunjukkan jumlah permintaan kapasitas yang telah terpenuhi. Untuk melihat ID instans di Armada Spot, pilih panah luaskan, atau pilih armada dan pilih instans.
4. Untuk Armada Spot, Kapasitas menunjukkan berapa banyak kapasitas yang diminta terpenuhi. Untuk melihat ID instans di Armada Spot, pilih ID armada untuk membuka halaman detailnya dan cari panel Instans.

AWS CLI

Untuk menemukan Instans Spot untuk permintaan

Gunakan perintah describe-spot-instance-quests berikut.

```
aws ec2 describe-spot-instance-requests \
    --spot-instance-request-ids sir-0e54a519c9EXAMPLE \
    --query "SpotInstanceRequests[*].{ID:InstanceId}"
```

Berikut ini adalah output contoh:

```
[
    {
        "ID": "i-1234567890abcdef0"
    },
    {
        "ID": "i-0598c7d356eba48d7"
    }
]
```

PowerShell

Untuk menemukan Instans Spot untuk permintaan

Gunakan Get-EC2SpotInstanceRequest cmdlet.

```
(Get-EC2SpotInstanceRequest -SpotInstanceRequestId sir-0e54a519c9EXAMPLE).InstanceId
```

### Menghentikan Instans Spot

Jika Anda tidak memerlukan Instans Spot sekarang, tetapi ingin memulai ulang nanti tanpa kehilangan data yang disimpan di volume Amazon EBS, Anda dapat menghentikannya. Langkah-langkah untuk menghentikan Instance Spot mirip dengan langkah-langkah untuk menghentikan On-Demand Instance.

> **Note**
>
> Saat Instans Spot dihentikan, Anda dapat mengubah beberapa atribut instansnya, tetapi tidak jenis instansnya.
> Kami tidak mengenakan biaya penggunaan untuk Instans Spot yang dihentikan, atau biaya transfer data, tetapi kami mengenakan biaya penyimpanan untuk setiap volume Amazon EBS.

### Batasan

- Anda hanya dapat menghentikan Instans Spot jika Instans Spot diluncurkan dari Permintaan Instans Spot `persistent`.
- Anda tidak dapat menghentikan Instans Spot jika permintaan Instans Spot yang terkait dibatalkan. Ketika permintaan Instans Spot dibatalkan, Anda hanya dapat mengakhiri Instans Spot.

- Anda tidak dapat menghentikan Instans Spot jika instans itu adalah bagian dari armada atau grup peluncuran, atau grup Zona Ketersediaan.

Console

Untuk menghentikan Instance Spot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih Instance Spot. Jika Anda tidak menyimpan ID instance dari Instance Spot, lihatthe section called “Menemukan Instans Spot Anda”.
4. Pilih Status instans, Hentikan instans.
5. Ketika diminta konfirmasi, pilih Berhenti.

AWS CLI

Untuk menghentikan Instance Spot

Gunakan perintah stop-instance untuk menghentikan Instans Spot secara manual.

```
aws ec2 stop-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk menghentikan Instance Spot

Gunakan Stop-EC2Instance cmdlet.

```
Stop-EC2Instance -InstanceId i-1234567890abcdef0
```

Memulai Instans Spot

Anda dapat memulai Instans Spot yang sebelumnya Anda hentikan.

Prasyarat

Anda hanya dapat memulai Instans Spot jika:

- Anda menghentikan Instans Spot secara manual.

- Instance Spot adalah sebuah EBS-backed instance.
- Kapasitas Instans Spot tersedia.
- Harga Spot lebih rendah dari harga maksimum Anda.

Batasan

- Anda tidak dapat memulai Instans Spot jika instans itu adalah bagian dari armada atau grup peluncuran, atau grup Zona Ketersediaan.

Langkah-langkah untuk memulai Instans Spot mirip dengan langkah-langkah untuk memulai On-Demand Instance.

Console

Untuk memulai Instans Spot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih Instance Spot. Jika Anda tidak menyimpan ID instance dari Instance Spot, lihatthe section called “Menemukan Instans Spot Anda”.
4. Pilih Status instans, Mulai instans.

AWS CLI

Untuk memulai Instans Spot

Gunakan perintah start-instance untuk memulai Instans Spot secara manual.

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk memulai Instans Spot

Gunakan Start-EC2Instance cmdlet.

```
Start-EC2Instance -InstanceId i-1234567890abcdef0
```

Menghentikan Instans Spot

> **Warning**
>
> Mengakhiri instance bersifat permanen dan tidak dapat diubah.
> Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
> Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Jika Anda mengakhiri Instans Spot yang sedang berjalan atau berhenti yang diluncurkan oleh permintaan Spot persisten, permintaan Instans Spot akan beralih ke status `open` sehingga Instans Spot baru dapat diluncurkan. Untuk memastikan bahwa tidak ada instans Spot baru yang diluncurkan, maka Anda harus terlebih dahulu membatalkan permintaan Instans Spot.

Jika Anda membatalkan permintaan Instans Spot `active` yang memiliki Instans Spot berjalan, maka Instans Spot yang berjalan itu tidak akan berhenti secara otomatis; Anda harus secara manual mengakhiri Instans Spot tersebut.

Jika Anda membatalkan permintaan Instans Spot `disabled` yang memiliki Instans Spot yang berhenti, maka Instans Spot yang berhenti akan secara otomatis diakhiri oleh layanan Amazon EC2 Spot. Mungkin ada jeda pendek antara saat Anda membatalkan permintaan Instans Spot dan ketika layanan Spot mengakhiri Instans Spot.

Untuk informasi selengkapnya, lihat Membatalkan permintaan Instans Spot.

Console

Untuk mengakhiri Instans Spot secara manual

1. Sebelum Anda mengakhiri sebuah instans, pastikan bahwa Anda tidak akan kehilangan data apa pun dengan memeriksa apakah volume Amazon EBS Anda tidak akan dihapus pada saat pengakhiran, dan apakah Anda telah menyalin semua data yang Anda perlukan dari volume penyimpanan instans Anda ke penyimpanan persisten, seperti sebagai Amazon EBS atau Amazon S3.

2. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
3. Di panel navigasi, pilih Instans.
4. Pilih Instance Spot. Jika Anda tidak menyimpan ID instance dari Instance Spot, lihatthe section called “Menemukan Instans Spot Anda”.
5. Pilih Status Instance, Hentikan (hapus) instance.
6. Pilih Hentikan (hapus) saat diminta konfirmasi.

AWS CLI

Untuk mengakhiri Instans Spot secara manual

Gunakan perintah terminate-instance untuk mengakhiri Instans Spot secara manual.

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7
```

PowerShell

Untuk mengakhiri Instans Spot secara manual

Gunakan Remove-EC2Instance cmdlet.

```
Remove-EC2Instance -InstanceId i-1234567890abcdef0
```

## Interupsi Instans Spot

Anda dapat meluncurkan Instans Spot pada kapasitas EC2 cadangan untuk mendapatkan diskon besar, dengan syarat mengembalikan kapasitas itu saat Amazon EC2 membutuhkannya kembali. Saat Amazon EC2 mengklaim kembali Instans Spot, kami menyebut peristiwa ini sebagai interupsi Instans Spot.

Permintaan untuk Instans Spot dapat sangat bervariasi dari waktu ke waktu, dan ketersediaan Instans Spot juga dapat sangat bervariasi tergantung pada berapa banyak instans EC2 yang tidak terpakai yang tersedia. Selalu ada kemungkinan Instans Spot Anda akan diinterupsi. Berikut ini adalah kemungkinan alasan Amazon EC2 menginterupsi Instans Spot Anda:

Kapasitas

Amazon EC2 dapat mengganggu menginterupsi Instans Spot Anda saat membutuhkannya kembali. EC2 mengeklaim kembali instans Anda terutama untuk menggunakan kembali

kapasitas, tetapi dapat juga terjadi karena alasan lain seperti pemeliharaan host atau penghentian penggunaan perangkat keras.

Harga

Harga Spot lebih tinggi dari harga maksimum Anda.

Anda dapat menentukan harga maksimum dalam permintaan Spot Anda. Jika Anda menentukan harga maksimum, instans Anda akan lebih sering diinterupsi daripada jika Anda memilih untuk tidak nenentukannya.

Batasan

Jika permintaan Spot Anda menyertakan batasan seperti grup peluncuran atau grup Zona Ketersediaan, Instans Spot diakhiri sebagai grup saat batasan tidak dapat lagi dipenuhi.

Saat Amazon EC2 menyela Instans Spot, instans akan dihentikan, dihentikan, atau hibernasi, tergantung pada perilaku interupsi yang Anda tentukan saat membuat permintaan Spot.

Daftar Isi


- Perilaku interupsi Instance Spot
- Bersiaplah untuk interupsi Instance Spot
- Memulai interupsi Instans Spot
- Pemberitahuan interupsi Instans Spot
- Menemukan Instans Spot yang diinterupsi
- Menentukan apakah Amazon EC2 mengakhiri Instans Spot
- Penagihan untuk Instans Spot yang diinterupsi


## Perilaku interupsi Instance Spot

Saat membuat permintaan Spot, Anda dapat menentukan perilaku interupsi. Berikut ini adalah kemungkinan perilaku interupsi:

- Menghentikan
- Hibernasi
- Mengakhiri

Perilaku defaultnya adalah Amazon EC2 mengakhiri Instans Spot saat instans tersebut terganggu.

### Menghentikan Instans Spot yang terinterupsi

Anda dapat menentukan bahwa Amazon EC2 akan menghentikan Instans Spot Anda ketika instans diinterupsi. Tipe permintaan Instans Spot harus `persistent`. Anda tidak dapat menentukan grup peluncuran dalam permintaan Instans Spot. Untuk Armada EC2 atau Armada Spot, jenis permintaan harus`maintain`.

#### Pertimbangan-pertimbangan

- Hanya Amazon EC2 yang dapat memulai ulang Instans Spot yang berhenti karena diinterupsi.
- Untuk Instans Spot yang diluncurkan oleh permintaan Instans Spot `persistent`: Amazon EC2 memulai ulang instans yang dihentikan ketika kapasitas tersedia di Zona Ketersediaan yang sama dan untuk tipe instans yang sama dengan instans yang dihentikan (spesifikasi peluncuran yang sama harus digunakan).
- Saat Instans Spot dihentikan, Anda dapat memodifikasi beberapa atribut instans, tetapi tidak untuk tipe instansnya. Jika Anda melepaskan atau menghapus volume EBS, volume tersebut tidak akan dilampirkan saat Instans Spot dimulai. Jika Anda melepaskan volume root dan Amazon EC2 mencoba memulai Instans Spot, instans akan gagal dimulai dan Amazon EC2 akan mengakhiri instans yang berhenti.
- Anda dapat mengakhiri Instans Spot saat instans berhenti.
- Jika Anda membatalkan permintaan Instans Spot, Armada EC2, atau Armada Spot, Amazon EC2 akan mengakhiri semua Instans Spot terkait yang berhenti.
- Saat Instans Spot yang diinterupsi dihentikan, Anda hanya dikenai biaya untuk volume EBS, yang dipertahankan. Dengan Armada EC2 dan Armada Spot, jika Anda memiliki banyak instans yang dihentikan, Anda dapat melebihi batas jumlah volume EBS untuk akun Anda. Untuk informasi selengkapnya tentang cara penagihan saat Instans Spot diinterupsi, lihat Penagihan untuk Instans Spot yang diinterupsi.
- Pastikan Anda terbiasa dengan implikasi berhentinya sebuah instans. Untuk informasi tentang apa yang terjadi saat sebuah instans berhenti, lihat Perbedaan antara status instance.

### Menghibernasi Instans Spot yang diinterupsi

Anda dapat menentukan bahwa Amazon EC2 akan menghibernasi Instans Spot Anda ketika instans diinterupsi. Untuk informasi selengkapnya, lihat Hibernasi instans Amazon EC2 Anda.

Amazon EC2 sekarang menawarkan pengalaman hibernasi yang sama untuk Instans Spot seperti yang saat ini tersedia untuk On-Demand Instans. Layanan ini menawarkan dukungan yang lebih luas, yang sekarang mendukung hal-hal berikut ini untuk hibernasi Instans Spot:

- Lebih banyak AMI yang didukung
- Lebih banyak keluarga instans yang didukung
- User-initiated hibernasi

### Mengakhiri Instans Spot yang diinterupsi

Saat menginterupsi Instans Spot, Amazon EC2 mengakhiri instans secara default, kecuali Anda menentukan perilaku interupsi berbeda, seperti berhenti atau hibernasi. Untuk informasi selengkapnya, lihat Menghentikan instans Amazon EC2.

### Bersiaplah untuk interupsi Instance Spot

Permintaan untuk Instans Spot dapat sangat bervariasi dari waktu ke waktu, dan ketersediaan Instans Spot juga dapat sangat bervariasi tergantung pada berapa banyak instans EC2 tidak terpakai yang tersedia. Selalu ada kemungkinan Instans Spot Anda akan diinterupsi. Oleh karena itu, Anda harus memastikan bahwa aplikasi Anda siap menghadapi interupsi Instans Spot.

Kami merekomendasikan Anda untuk mengikuti praktik terbaik ini sehingga Anda siap menghadapi interupsi Instans Spot.

- Buat permintaan Spot menggunakan grup Auto Scaling. Jika Instans Spot Anda diinterupsi, grup Auto Scaling akan secara otomatis meluncurkan instans pengganti. Untuk informasi selengkapnya, lihat Grup Auto Scaling dengan beberapa tipe instans dan opsi pembelian dalam Panduan Pengguna Amazon EC2 Auto Scaling.
- Pastikan instans Anda siap digunakan segera setelah permintaan dipenuhi dengan menggunakan Amazon Machine Image (AMI) yang berisi konfigurasi perangkat lunak yang diperlukan. Anda juga dapat menggunakan data pengguna untuk menjalankan perintah saat memulai.
- Data pada volume penyimpanan instans hilang saat instans dihentikan atau diakhiri. Cadangkan semua data penting pada volume penyimpanan instans ke penyimpanan yang lebih persisten, seperti Amazon S3, Amazon EBS, atau Amazon DynamoDB.
- Simpan data penting secara teratur di tempat yang tidak terpengaruh jika Instans Spot diakhiri. Misalnya, Anda dapat menggunakan Amazon S3, Amazon EBS, atau DynamoDB.

- Bagilah pekerjaan menjadi tugas-tugas kecil (menggunakan Grid, Hadoop, atau arsitektur berbasis antrean) atau gunakan titik pemeriksaan sehingga Anda dapat sering menyimpan pekerjaan.
- Amazon EC2 memancarkan sinyal rekomendasi penyeimbangan kembali ke Instans Spot saat instans berisiko tinggi mengalami interupsi. Anda dapat mengandalkan rekomendasi penyeimbangan kembali untuk secara proaktif mengelola interupsi Instans Spot tanpa harus menunggu pemberitahuan interupsi Instans Spot dua menit. Untuk informasi selengkapnya, lihat Rekomendasi penyeimbangan ulang instans EC2.
- Gunakan pemberitahuan interupsi Instans Spot dua menit untuk memantau status Instans Spot Anda. Untuk informasi selengkapnya, lihat Pemberitahuan interupsi Instans Spot.
- Meskipun kami berusaha semaksimal mungkin untuk memberikan peringatan ini, ada kemungkinan Instans Spot Anda diinterupsi sebelum peringatan tersebut datang. Uji aplikasi Anda untuk memastikan bahwa aplikasi tersebut menangani interupsi instans yang tidak terduga dengan baik, meskipun Anda memantau sinyal rekomendasi penyeimbangan kembali dan pemberitahuan interupsi. Anda dapat melakukan ini dengan menjalankan aplikasi menggunakan On-Demand Instance dan kemudian menghentikan On-Demand Instance sendiri.
- Jalankan eksperimen injeksi kesalahan terkontrol dengan AWS Fault Injection Service untuk menguji bagaimana aplikasi Anda merespons saat Instans Spot Anda terganggu. Untuk informasi selengkapnya, lihat Tutorial: Uji interupsi Instans Spot menggunakan AWS FIS dalam di Panduan Pengguna AWS Fault Injection Service .

### Memulai interupsi Instans Spot

Anda dapat memilih permintaan Instans Spot atau permintaan Armada Spot di konsol Amazon EC2 dan memulai interupsi sehingga Anda dapat menguji bagaimana aplikasi di Instans Spot menangani interupsi. Saat Anda memulai interupsi Instans Spot, Amazon EC2 memberi tahu Anda bahwa Instans Spot Anda akan diinterupsi dalam dua menit, dan kemudian, setelah dua menit, instans akan diinterupsi.

Layanan dasar yang melakukan interupsi Instance Spot adalah AWS Fault Injection Service (AWS FIS). Untuk informasi tentang AWS FIS, lihat AWS Fault Injection Service.

> **Note**
>
> Perilaku interupsi adalah `terminate`, `stop`, dan `hibernate`. Jika Anda mengatur perilaku interupsi ke `hibernate`, saat Anda memulai interupsi Instans Spot, proses hibernasi akan segera dimulai.

Memulai gangguan Instans Spot didukung di semua Wilayah AWS kecuali Asia Pasifik (Jakarta), Asia Pasifik (Osaka), China (Beijing), China (Ningxia), dan Timur Tengah (UEA).

Daftar Isi


- Memulai interupsi Instans Spot
- Verifikasi interupsi Instans Spot
- Kuota


## Memulai interupsi Instans Spot

Anda dapat menggunakan konsol EC2 untuk memulai interupsi Instans Spot dengan cepat. Ketika Anda memilih permintaan Instans Spot, Anda dapat memulai interupsi satu Instans Spot. Ketika Anda memilih permintaan Armada Spot, Anda dapat memulai interupsi banyak Instans Spot sekaligus.

Untuk eksperimen lanjutan untuk menguji interupsi Instance Spot, Anda dapat membuat eksperimen sendiri menggunakan AWS FIS konsol.

Untuk memulai interupsi satu Instans Spot dalam satu permintaan Instans Spot menggunakan konsol EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Instans Spot, lalu pilih Tindakan, Mulai interupsi. Anda tidak dapat memilih banyak permintaan Instans Spot untuk memulai interupsi.
4. Di kotak dialog Mulai interupsi Instans Spot, pada Akses layanan, gunakan peran default, atau pilih peran yang sudah ada. Untuk memilih peran yang sudah ada, pilih Gunakan peran layanan yang ada, lalu untuk Peran IAM, pilih peran yang akan digunakan.
5. Saat Anda siap untuk memulai interupsi Instans Spot, pilih Mulai interupsi.

Untuk memulai interupsi satu atau lebih Instans Spot dalam satu permintaan Armada Spot menggunakan konsol EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot, lalu pilih Tindakan, Mulai interupsi. Anda tidak dapat memilih banyak permintaan Armada Spot untuk memulai interupsi.

4. Dalam kotak dialog Tentukan jumlah Instans Spot, untuk Jumlah instans yang akan diinterupsi, masukkan jumlah Instans Spot yang akan diinterupsi, lalu pilih Konfirmasi.

> **Note**
>
> Jumlah tersebut tidak boleh melebihi jumlah Instans Spot dalam armada atau kuota Anda untuk jumlah Instans Spot yang AWS FIS dapat mengganggu per percobaan.

5. Di kotak dialog Mulai interupsi Instans Spot, pada Akses layanan, gunakan peran default, atau pilih peran yang sudah ada. Untuk memilih peran yang sudah ada, pilih Gunakan peran layanan yang ada, lalu untuk Peran IAM, pilih peran yang akan digunakan.
6. Saat Anda siap untuk memulai interupsi Instans Spot, pilih Mulai interupsi.

Untuk membuat eksperimen yang lebih canggih untuk menguji interupsi Instance Spot menggunakan AWS FIS konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari panel navigasi, pilih Permintaan Spot.
3. Pilih Tindakan, Buat eksperimen lanjutan.

   Kon AWS FIS sol terbuka. Untuk informasi selengkapnya, lihat Tutorial: Uji interupsi Instans Spot menggunakan AWS FIS dalam di Panduan Pengguna AWS Fault Injection Service .

Verifikasi interupsi Instans Spot

Setelah Anda memulai interupsi, berikut ini yang akan terjadi:

- Instans Spot menerima rekomendasi penyeimbangan kembali instans.
- Pember itahuan gangguan Instans Spot dikeluarkan dua menit sebelum instans AWS FIS Anda interupsi.
- Setelah dua menit, Instans Spot akan diinterupsi.
- Instans Spot yang dihentikan oleh AWS FIS tetap dihentikan sampai Anda memulai ulang.

Untuk memverifikasi bahwa instans diinterupsi setelah Anda memulai interupsi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Dari panel navigasi, buka Permintaan Spot dan Instans di tab atau jendela peramban yang terpisah.
3. Untuk Permintaan Spot, pilih permintaan Instans Spot atau permintaan Armada Spot. Status awal adalah `fulfilled`. Setelah instans diinterupsi, status berubah sebagai berikut, tergantung pada perilaku interupsi:
   - `terminate` – Status berubah menjadi `instance-terminated-by-experiment`.
   - `stop` – Status berubah menjadi `marked-for-stop-by-experiment`, kemudian `instance-stopped-by-experiment`.
4. Untuk Instans, pilih Instans Spot. Status awal adalah `Running`. Dua menit setelah Anda menerima pemberitahuan diinterupsi Instans Spot, status berubah sebagai berikut, tergantung pada perilaku interupsi:
   - `stop` – Status berubah menjadi `Stopping`, kemudian `Stopped`.
   - `terminate` – Status berubah menjadi `Shutting-down`, kemudian `Terminated`.

Kuota

Anda Akun AWS memiliki kuota default berikut untuk jumlah Instans Spot yang AWS FIS dapat mengganggu per percobaan.

| Nama | Default | Dapat disesuaikan | Deskripsi |
| --- | --- | --- | --- |
| Target SpotInstances untuk aws:ec2:send-spot-instance- | Setiap Wilayah yang didukung: 5 | Ya | Jumlah maksimum Instans Spot yang dapat ditargetkan aws:ec2:send-spot-instance-interruptions saat Anda mengidentifikasi target menggunakan tanda, per eksperimen. |

Anda dapat meminta penambahan kuota. Untuk informasi selengkapnya, lihat Meminta peningkatan kuota di Panduan Pengguna Kuota Layanan.

Untuk melihat semua kuota untuk AWS FIS, buka konsol Kuota Layanan. Pada panel navigasi, pilih Layanan AWS dan pilih AWS Fault Injection Service. Anda juga dapat melihat semua kuota untuk AWS Fault Injection Service di Panduan Pengguna AWS Fault Injection Service .

Pemberitahuan interupsi Instans Spot

Pemberitahuan interupsi Instans Spot adalah peringatan yang dikeluarkan dua menit sebelum Amazon EC2 menghentikan atau mengakhiri Instans Spot Anda. Jika Anda menentukan hibernasi sebagai perilaku interupsi, Anda akan menerima pemberitahuan interupsi, tetapi Anda tidak menerima peringatan dua menit karena proses hibernasi langsung dimulai.

Cara terbaik agar Anda dapat menangani interupsi Instans Spot dengan baik adalah dengan merancang aplikasi Anda agar toleran terhadap kesalahan. Untuk melakukannya, Anda dapat memanfaatkan pemberitahuan interupsi Instans Spot. Kami menyarankan Anda untuk memeriksa pemberitahuan interupsi ini setiap 5 detik.

Pemberitahuan interupsi tersedia sebagai EventBridge peristiwa dan sebagai item dalam metadata instans di Instans Spot. Pemberitahuan interupsi dipancarkan dengan upaya yang terbaik.

Peringatan Interupsi Instans Spot EC2 kejadian

Saat akan menginterupsi Instans Spot Anda, Amazon EC2 memancarkan peristiwa dua menit sebelum interupsi sebenarnya (kecuali untuk hibernasi, yang mendapatkan pemberitahuan interupsi, tetapi tidak dua menit sebelumnya, karena hibernasi langsung dimulai). Peristiwa ini dapat dideteksi oleh Amazon EventBridge. Untuk informasi selengkap EventBridge nya tentang acara, lihat Panduan EventBridge Pengguna Amazon. Untuk contoh terperinci yang akan memandu Anda tentang cara membuat dan menggunakan aturan peristiwa, lihat Memanfaatkan Notifikasi Interupsi Instans Spot Amazon EC2.

Berikut ini adalah contoh peristiwa untuk interupsi Instans Spot. Nilai yang mungkin untuk `instance-action` adalah `hibernate`, `stop`, dan `terminate`.

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "EC2 Spot Instance Interruption Warning",
    "source": "aws.ec2",
    "account": "123456789012",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2a:instance/i-1234567890abcdef0"],
```

```
    "detail": {
        "instance-id": "i-1234567890abcdef0",
        "instance-action": "action"
    }
}
```

> **Note**
>
> Format ARN dari peristiwa interupsi Spot Instance adalah`arn:aws:ec2:`*availability-zone*`:instance/`*instance-id*. Format ini berbeda dari format ARN sumber daya EC2.

instance-action

Item `instance-action` menentukan tindakan dan perkiraan waktu, dalam UTC, kapan tindakan akan terjadi.

Jika Instans Spot Anda ditandai untuk dihentikan atau diakhiri oleh Amazon EC2, item `instance-action` akan ada di metadata instans Anda. Jika tidak, item itu tidak ada. Anda dapat mengambil `instance-action` menggunakan Instance Metadata Service Version 2 (IMDSv2) sebagai berikut.

Linux

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/spot/instance-action
```

Windows

```
[string]$token = Invoke-RestMethod `
    -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
    -Method PUT -Uri http://169.254.169.254/latest/meta-data/spot/instance-action
```

Contoh output berikut menunjukkan waktu saat instans ini akan dihentikan.

```
{"action": "stop", "time": "2017-09-18T08:22:00Z"}
```

Output contoh berikut menunjukkan waktu saat instans ini akan diakhiri.

```
{"action": "terminate", "time": "2017-09-18T08:22:00Z"}
```

Jika Amazon EC2 tidak bersiap untuk menghentikan atau mengakhiri instans, atau jika Anda sendiri mengakhiri instans, `instance-action` tidak ada dalam metadata instans dan Anda menerima kesalahan HTTP 404 saat Anda mencoba untuk mengambilnya kembali.

waktu penghentian

I `termination-time` tem menentukan perkiraan waktu dalam UTC ketika instance akan menerima sinyal shutdown.

> **Note**
>
> Item ini dipertahankan untuk kompatibilitas mundur; Anda seharusnya menggunakan `instance-action`.

Jika Instans Spot Anda ditandai untuk dihentikan oleh Amazon EC2 (baik karena gangguan Instans Spot di mana perilaku interupsi disetel `keterminate`, atau karena pembatalan permintaan Instans Spot persisten), `termination-time` item tersebut ada dalam metadata ??? instans Anda. Jika tidak, item itu tidak ada. Anda dapat mengambil `termination-time` menggunakan IMDSv2 sebagai berikut.

Linux

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`
if curl -H "X-aws-ec2-metadata-token: $TOKEN" -s http://169.254.169.254/latest/meta-data/spot/termination-time | grep -q .*T.*Z; then echo termination_scheduled; fi
```

Windows

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/meta-data/spot/termination-time
```

Berikut ini adalah output contoh.

```
2015-01-05T18:02:00Z
```

Jika Amazon EC2 tidak bersiap untuk menghentikan instans (baik karena tidak ada gangguan Instans Spot atau karena perilaku interupsi Anda disetel ke `stop` atau`hibernate`), atau jika Anda menghentikan Instans Spot sendiri, `termination-time` item tersebut tidak ada dalam metadata instans (sehingga Anda menerima kesalahan HTTP 404) atau berisi nilai yang bukan nilai waktu.

Jika Amazon EC2 gagal untuk mengakhiri instans, status permintaan diatur ke `fulfilled`. Nilai `termination-time` tetap dalam metadata instans dengan perkiraan waktu semula, yang sekarang sudah berlalu.

Menemukan Instans Spot yang diinterupsi

Saat Anda menjelaskan instans EC2, hasilnya termasuk Instans Spot Anda. Siklus hidup instans dari instans Spot adalah `spot`. Status instans dari Instans Spot bisa berupa `stopped` atau `terminated`, tergantung pada perilaku interupsi yang Anda konfigurasikan. Untuk instans Spot hibernasi, status instans adalah `stopped`.

Untuk detail tambahan tentang alasan interupsi, periksa kode status permintaan Spot. Untuk informasi selengkapnya, lihat the section called “Mendapatkan status permintaan Instance Spot”.

Console

Untuk menemukan Instans Spot yang terputus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Terapkan filter berikut: Siklus hidup instans=spot.
4. Terapkan filter Status instans=berhenti atau Status instans=diakhiri tergantung pada perilaku interupsi yang Anda konfigurasikan.
5. Untuk setiap Instans Spot, di tab Detail, pada Detail instans, temukan Pesan transisi status. Kode berikut menunjukkan bahwa Instans Spot diinterupsi.

   - `Server.SpotInstanceShutdown`
   - `Server.SpotInstanceTermination`

AWS CLI

Untuk menemukan Instans Spot yang terputus

Gunakan perintah describe-instance dengan opsi. `--filters` Untuk mencantumkan hanya ID instance dalam output, sertakan `--query` opsi.

Jika perilaku interupsi adalah menghentikan Instans Spot, gunakan contoh berikut:

```
aws ec2 describe-instances \
    --filters Name=instance-lifecycle,Values=spot \
              Name=instance-state-name,Values=terminated \
              Name=state-reason-code,Values=Server.SpotInstanceTermination \
    --query "Reservations[*].Instances[*].InstanceId"
```

Jika perilaku interupsi adalah menghentikan Instans Spot, gunakan contoh berikut:

```
aws ec2 describe-instances \
    --filters Name=instance-lifecycle,Values=spot \
              Name=instance-state-name,Values=stopped \
              Name=state-reason-code,Values=Server.SpotInstanceShutdown \
    --query "Reservations[*].Instances[*].InstanceId"
```

PowerShell

Untuk menemukan Instans Spot yang terputus

Gunakan Get-EC2Instance cmdlet.

Jika perilaku interupsi adalah menghentikan Instans Spot, gunakan contoh berikut:

```
(Get-EC2Instance `
    -Filter @{Key="instance-lifecycle"; Values="spot"} `
            @{Key="instance-state-name"; Values="terminated"} `
            @{Key="state-reason-code";
 Values="Server.SpotInstanceTermination"}).Instances.InstanceId
```

Jika perilaku interupsi adalah menghentikan Instans Spot, gunakan contoh berikut:

```
(Get-EC2Instance `
    -Filter @{Key="instance-lifecycle"; Values="spot"} `
            @{Key="instance-state-name"; Values="stopped"} `
```

```
            @{Key="state-reason-code";
Values="Server.SpotInstanceTermination"}).Instances.InstanceId
```

Menentukan apakah Amazon EC2 mengakhiri Instans Spot

Instans Spot berjalan hingga Amazon EC2 menghentikannya sebagai respons terhadap gangguan Instans Spot, atau sampai Anda menghentikannya sendiri. Untuk informasi selengkapnya, lihat the section called “Perilaku interupsi”.

Setelah Instans Spot dihentikan, Anda dapat menggunakannya AWS CloudTrail untuk melihat apakah Amazon EC2 menghentikannya. Jika CloudTrail log menyertakan a`BidEvictedEvent`, ini menunjukkan bahwa Amazon EC2 menghentikan Instans Spot. Jika sebaliknya Anda melihat `TerminateInstances` peristiwa, ini menunjukkan bahwa pengguna menghentikan Instance Spot.

Atau, jika Anda ingin menerima pemberitahuan bahwa Amazon EC2 akan mengganggu Instans Spot Anda, gunakan Amazon EventBridge untuk menanggapi peristiwa Peringatan Interupsi Instans Spot EC2.

Untuk melihat BidEvictedEvent acara di CloudTrail

1. Buka CloudTrail konsol di https://console.aws.amazon.com/cloudtrail/.
2. Pada panel navigasi, pilih Riwayat peristiwa.
3. Dari daftar filter, pilih Nama acara, dan kemudian di bidang filter di sebelah kanan, masukkan**`BidEvictedEvent`**.
4. (Opsional) Pilih rentang waktu.
5. Jika daftar tidak kosong, pilih BidEvictedEvent dari entri yang dihasilkan untuk membuka halaman detailnya. Anda dapat menemukan informasi tentang Instans Spot di panel catatan peristiwa, termasuk ID Instans Spot. Berikut ini adalah contoh catatan peristiwa.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "accountId": "123456789012",
        "invokedBy": "ec2.amazonaws.com"
    },
    "eventTime": "2016-08-16T22:30:00Z",
    "eventSource": "ec2.amazonaws.com",
    "userAgent": "ec2.amazonaws.com",
    "sourceIPAddress": "ec2.amazonaws.com",
```

```
    "eventName": "BidEvictedEvent",
    "awsRegion": "us-east-2",
    "eventID": "d27a6096-807b-4bd0-8c20-a33a83375054",
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "123456789012",
    "RequestParameters": null,
    "ResponseElements": null,
    "serviceEventDetails": {
        "instanceIdSet": [
          "i-1eb2ac8eEXAMPLE"
        ]
    }
}
```

6. Jika Anda tidak menemukan entri untuk `BidEvictedEvent` acara tersebut, masukkan **`TerminateInstances`** sebagai nama acara. Untuk informasi selengkapnya tentang catatan acara untuk`TerminateInstances`, lihatthe section called “Contoh acara API Amazon EC2”.

Penagihan untuk Instans Spot yang diinterupsi

Ketika Instans Spot terganggu, Anda akan dikenakan biaya untuk penggunaan volume misalnya dan EBS, dan Anda mungkin dikenakan biaya lain, sebagai berikut.

Penggunaan instans

| Siapa yang menginterupsi Instans Spot | Sistem operasi | Interupsi dalam satu jam pertama | Interupsi dalam berapa pun jam setelah satu jam pertama |
|---|---|---|---|
| Jika Anda menghentikan atau mengakhiri Instans Spot | Windows dan Linux (tidak termasuk SUSE) | Dikenai biaya untuk detik yang digunakan | Dikenai biaya untuk detik yang digunakan |
| | SEGAR | Dikenai biaya selama satu jam penuh meskipun Anda menggunakan sebagian jam | Dikenai biaya selama satu jam penuh yang digunakan, dan dikenai biaya untuk sebagian jam yang diinterupsi |

| Siapa yang menginterupsi Instans Spot | Sistem operasi | Interupsi dalam satu jam pertama | Interupsi dalam berapa pun jam setelah satu jam pertama |
|---|---|---|---|
| Jika Amazon EC2 menginterupsi Instans Spot | Windows dan Linux (tidak termasuk SUSE) | Tidak dikenai biaya | Dikenai biaya untuk detik yang digunakan |
| | SEGAR | Tidak dikenai biaya | Dikenai biaya selama satu jam penuh yang digunakan, tetapi tidak dikenai biaya untuk sebagian jam yang diinterupsi |

Penggunaan volume EBS

Saat Instans Spot yang diinterupsi dihentikan, Anda hanya dikenai biaya untuk volume EBS, yang dipertahankan.

Dengan Armada EC2 dan Armada Spot, jika Anda memiliki banyak instans yang dihentikan, Anda dapat melebihi batas jumlah volume EBS untuk akun Anda.

## Rekomendasi penyeimbangan ulang instans EC2

Rekomendasi penyeimbangan kembali instans EC2 adalah sinyal yang memberi tahu Anda saat Instans Spot berada pada risiko interupsi yang tinggi. Sinyal dapat tiba lebih cepat daripada pemberitahuan interupsi Instans Spot dua menit, yang memberi Anda kesempatan untuk mengelola Instans Spot secara proaktif. Anda dapat memutuskan untuk menyeimbangkan kembali beban kerja Anda ke Instans Spot baru atau lama yang tidak berisiko tinggi mengalami interupsi.

Amazon EC2 tidak selalu dapat mengirim sinyal rekomendasi penyeimbangan kembali sebelum pemberitahuan interupsi Instans Spot dua menit. Oleh karena itu, sinyal rekomendasi penyeimbangan kembali dapat tiba bersama dengan pemberitahuan interupsi dua menit.

Rekomendasi rebalance tersedia sebagai EventBridge peristiwa dan sebagai item dalam metadata ??? instans di Instans Spot. Peristiwa dipancarkan atas dasar upaya terbaik.

> **Note**
>
> Rekomendasi penyeimbangan kembali hanya didukung untuk Instans Spot yang diluncurkan setelah 5 November 2020 00:00 UTC.

Daftar Isi


- Menyeimbangkan kembali tindakan yang dapat Anda lakukan
- Pantau sinyal rekomendasi penyeimbangan kembali
- Layanan yang menggunakan sinyal rekomendasi penyeimbangan kembali


## Menyeimbangkan kembali tindakan yang dapat Anda lakukan

Berikut adalah beberapa kemungkinan tindakan penyeimbangan ulang yang dapat Anda lakukan:

Pematian terkendali

Saat Anda menerima sinyal rekomendasi penyeimbangan ulang untuk Instans Spot, Anda dapat memulai prosedur pematian instans Anda, yang mungkin termasuk memastikan bahwa proses telah selesai sebelum menghentikannya. Misalnya, Anda dapat mengunggah log sistem atau aplikasi ke Amazon Simple Storage Service (Amazon S3), Anda dapat mematikan pekerja Amazon SQS, atau Anda dapat menyelesaikan penghapusan pendaftaran dari Sistem Nama Domain (DNS). Anda juga dapat menyimpan pekerjaan Anda di penyimpanan eksternal dan melanjutkannya di lain waktu.

Mencegah pekerjaan baru dijadwalkan

Saat Anda menerima sinyal rekomendasi penyeimbangan kembali untuk Instans Spot, Anda dapat mencegah pekerjaan baru dijadwalkan pada instans tersebut, sambil terus menggunakan instans tersebut hingga pekerjaan yang dijadwalkan selesai.

Luncurkan instans pengganti baru secara proaktif

Anda dapat mengonfigurasi grup Auto Scaling, Armada EC2, atau Armada Spot untuk secara otomatis meluncurkan Instans Spot pengganti ketika sinyal rekomendasi penyeimbangan kembali dipancarkan. Untuk informasi selengkapnya, lihat Menggunakan Penyeimbangan Kembali Kapasitas untuk menangani gangguan Amazon EC2 Spot di Panduan Pengguna Penskalaan Otomatis Amazon EC2, dan Gunakan Penyeimbangan Kembali Kapasitas di Armada EC2 dan Spot Fleet untuk menggantikan Instans Spot yang berisiko dalam panduan pengguna ini.

### Pantau sinyal rekomendasi penyeimbangan kembali

Anda dapat memantau sinyal rekomendasi penyeimbangan kembali sehingga Anda dapat mengambil tindakan yang ditentukan di bagian sebelumnya ketika sinyal dipancarkan. Sinyal rekomendasi rebalance tersedia sebagai peristiwa yang dikirim ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon Ev CloudWatch ents) dan sebagai metadata instans pada Instans Spot.

Pantau sinyal rekomendasi penyeimbangan kembali:

- Gunakan Amazon EventBridge
- Gunakan metadata instans

#### Gunakan Amazon EventBridge

Saat sinyal rekomendasi rebalance dipancarkan untuk Instans Spot, peristiwa untuk sinyal dikirim ke Amazon EventBridge. Jika EventBridge mendeteksi pola peristiwa yang cocok dengan pola yang ditentukan dalam aturan, EventBridge memanggil target (atau target) yang ditentukan dalam aturan.

Berikut adalah contoh peristiwa untuk sinyal rekomendasi penyeimbangan kembali.

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "EC2 Instance Rebalance Recommendation",
    "source": "aws.ec2",
    "account": "123456789012",
    "time": "yyyy-mm-ddThh:mm:ssZ",
    "region": "us-east-2",
    "resources": ["arn:aws:ec2:us-east-2:123456789012:instance/i-1234567890abcdef0"],
    "detail": {
        "instance-id": "i-1234567890abcdef0"
    }
}
```

Bidang berikut membentuk pola peristiwa yang ditentukan dalam aturan:

`"detail-type": "EC2 Instance Rebalance Recommendation"`

Mengidentifikasi bahwa peristiwa itu adalah peristiwa rekomendasi penyeimbangan kembali

`"source": "aws.ec2"`

Mengidentifikasi bahwa peristiwa tersebut itu dari Amazon EC2

### Buat EventBridge aturan

Anda dapat menulis EventBridge aturan dan mengotomatiskan tindakan apa yang harus diambil ketika pola peristiwa cocok dengan aturan.

Contoh berikut membuat EventBridge aturan untuk mengirim email, pesan teks, atau pemberitahuan push seluler setiap kali Amazon EC2 memancarkan sinyal rekomendasi rebalance. Sinyal dipancarkan sebagai peristiwa `EC2 Instance Rebalance Recommendation`, yang memicu tindakan yang ditentukan oleh aturan.

Sebelum membuat EventBridge aturan, Anda harus membuat topik Amazon SNS untuk email, pesan teks, atau pemberitahuan push seluler.

Untuk membuat EventBridge aturan untuk acara rekomendasi rebalance

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Pilih Buat aturan.
3. Untuk Tentukan detail aturan, lakukan hal berikut:
   a. Masukkan Nama untuk aturan tersebut dan, secara opsional, deskripsi.

      Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah yang sama dan di bus peristiwa yang sama.
   b. Untuk Bus peristiwa, pilih default. Saat layanan AWS di akun Anda membuat peristiwa, layanan tersebut akan selalu masuk ke bus peristiwa default akun.
   c. Untuk Tipe aturan, pilih Aturan dengan pola peristiwa.
   d. Pilih Selanjutnya.
4. Untuk Pola peristiwa build, lakukan hal berikut ini:
   a. Untuk Sumber acara, pilih AWS acara atau acara EventBridge mitra.
   b. Untuk Pola peristiwa, untuk contoh ini Anda akan menentukan pola peristiwa berikut agar cocok dengan peristiwa `EC2 Instance Rebalance Recommendation`, lalu pilih Simpan.

```
{
 "source": ["aws.ec2"],
 "detail-type": ["EC2 Instance Rebalance Recommendation"]
}
```

Untuk menambahkan pola peristiwa, Anda dapat menggunakan templat dengan memilih formulir pola peristiwa, atau menentukan pola Anda sendiri dengan memilih Pola kustom (editor JSON), sebagai berikut:

i. Untuk menggunakan templat untuk membuat pola peristiwa, lakukan hal berikut:

A. Pilih Formulir pola peristiwa.

B. Untuk Sumber peristiwa, pilih Layanan AWS .

C. Untuk Layanan AWS , pilih Armada Spot EC2.

D. Untuk Tipe peristiwa, pilih Rekomendasi Penyeimbangan Kembali Instans EC2.

E. Untuk menyesuaikan templat, pilih Edit pola dan buat perubahan Anda agar sesuai dengan contoh pola peristiwa.

ii. (Alternatif) Untuk menentukan pola peristiwa khusus, lakukan hal berikut:

A. Pilih Pola kustom (editor JSON).

B. Dalam kotak Pola peristiwa, tambahkan pola peristiwa untuk contoh ini.

c. Pilih Selanjutnya.

5. Untuk Pilih target, lakukan hal berikut:

a. Untuk Tipe Target, pilih Layanan AWS .

b. Untuk Memilih target, pilih topik SNS untuk mengirim email, pesan teks, atau notifikasi push seluler saat peristiwa tersebut terjadi.

c. Untuk Topik, pilih topik yang ada. Anda harus terlebih dahulu membuat topik Amazon SNS menggunakan konsol Amazon SNS. Untuk informasi selengkapnya, lihat Menggunakan Amazon SNS untuk pengiriman pesan aplikasi-ke-orang (A2P) dalam Panduan Developer Amazon Simple Notification Service.

d. (Opsional) Pada Pengaturan tambahan, Anda dapat mengonfigurasi pengaturan tambahan secara opsional. Untuk informasi selengkapnya, lihat Membuat EventBridge aturan Amazon yang bereaksi terhadap peristiwa (langkah 16) di Panduan EventBridge Pengguna Amazon.

e. Pilih Selanjutnya.

6. (Opsional) Untuk Tanda, Anda dapat secara opsional menetapkan satu atau beberapa tanda ke aturan, lalu pilih Berikutnya.

7. Untuk Tinjau dan buat, lakukan hal berikut:

a. Tinjau detail aturan dan modifikasi seperlunya.

b. Pilih Buat aturan.

Untuk informasi selengkapnya, lihat EventBridge aturan Amazon dan pola EventBridge acara Amazon di Panduan EventBridge Pengguna Amazon

Gunakan metadata instans

Kategori metadata instans `events/recommendations/rebalance` memberikan perkiraan waktu, dalam UTC, kapan sinyal rekomendasi penyeimbangan kembali dipancarkan untuk Instans Spot.

Kami menyarankan Anda untuk memeriksa sinyal rekomendasi penyeimbangan kembali setiap 5 detik agar Anda tidak melewatkan kesempatan untuk menjalankan rekomendasi penyeimbangan kembali.

Jika Instans Spot menerima rekomendasi penyeimbangan kembali, waktu sinyal dipancarkan ada dalam metadata instans. Anda dapat mengambil waktu saat sinyal itu dipancarkan sebagai berikut.

IMDSv2

Linux

Jalankan perintah berikut dari instance Linux Anda.

IMDSv2

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/recommendations/rebalance
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda

```
[string]$token = Invoke-RestMethod `
    -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
    -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod `
```

```
    -Headers @{"X-aws-ec2-metadata-token" = $token} `
    -Method GET -Uri http://169.254.169.254/latest/meta-data/events/recommendations/
rebalance
```

IMDSv1

Linux

Jalankan perintah berikut dari instance Linux Anda.

```
curl http://169.254.169.254/latest/meta-data/events/recommendations/rebalance
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda.

```
Invoke-RestMethod -Uri http://169.254.169.254/latest/meta-data/events/
recommendations/rebalance
```

Berikut ini adalah contoh output, yang menunjukkan waktu, dalam UTC, saat sinyal rekomendasi penyeimbangan kembali dipancarkan untuk Instans Spot.

```
{"noticeTime": "2020-10-27T08:22:00Z"}
```

Jika sinyal belum dipancarkan untuk instans itu, `events/recommendations/rebalance` tidak ada dan Anda akan menerima kesalahan HTTP 404 ketika Anda mencoba untuk mengambilnya kembali.

Layanan yang menggunakan sinyal rekomendasi penyeimbangan kembali

Amazon EC2 Auto Scaling, Armada EC2, dan Armada Spot menggunakan sinyal rekomendasi penyeimbangan kembali untuk memudahkan Anda mempertahankan ketersediaan beban kerja dengan secara proaktif menambah armada Anda dengan Instans Spot baru sebelum instans yang berjalan menerima pemberitahuan interupsi Instans Spot dua menit. Anda dapat meminta layanan ini untuk memantau dan secara proaktif merespons perubahan yang memengaruhi ketersediaan Instans Spot Anda. Untuk informasi selengkapnya, lihat berikut ini:

- Gunakan Penyeimbangan Kembali Kapasitas untuk menangani interupsi Amazon EC2 Spot di Panduan Pengguna Amazon EC2 Auto Scaling

- Gunakan Penyeimbangan Kembali Kapasitas di Armada EC2 dan Spot Fleet untuk menggantikan Instans Spot yang berisikodalam topik Armada dan Armada Spot EC2 di panduan pengguna ini

## Skor penempatan Spot

Fitur Skor penempatan Spot dapat merekomendasikan Wil AWS ayah, Zona Ketersediaan, atau Zona Lokal berdasarkan persyaratan kapasitas Spot Anda. Kapasitas spot berfluktuasi, dan Anda tidak dapat memastikan bahwa Anda akan selalu mendapatkan kapasitas yang Anda butuhkan. Skor penempatan Spot menunjukkan seberapa besar kemungkinan permintaan Spot akan berhasil di suatu Wilayah atau Zona Ketersediaan.

> **Note**
>
> Skor penempatan Spot tidak memberikan jaminan apa pun dalam hal kapasitas yang tersedia atau risiko interupsi. Skor penempatan Spot hanya berfungsi sebagai rekomendasi.

### Kasus penggunaan

Anda dapat menggunakan fitur skor penempatan Spot untuk hal-hal berikut:

- Untuk merelokasi dan menskalakan kapasitas komputasi Spot di Wilayah yang berbeda, sesuai kebutuhan, sebagai respons terhadap peningkatan kebutuhan kapasitas atau penurunan kapasitas yang tersedia di Wilayah saat ini.
- Untuk mengidentifikasi Zona Ketersediaan yang paling optimal untuk menjalankan beban kerja Zona Ketersediaan Tunggal.
- Untuk mensimulasikan kebutuhan kapasitas Spot di masa depan sehingga Anda dapat memilih Wilayah yang optimal untuk perluasan Spot-based beban kerja Anda.
- Untuk menemukan kombinasi tipe instans yang optimal untuk memenuhi kebutuhan kapasitas Spot Anda.

### Daftar Isi


- Batasan
- Biaya
- Cara kerja skor penempatan Spot
- Izin yang diperlukan untuk skor penempatan Spot

- Hitung skor penempatan Spot

Batasan

- Batas kapasitas target — Batas kapasitas target skor penempatan Spot Anda didasarkan pada penggunaan Spot terbaru Anda, sambil memperhitungkan potensi pertumbuhan penggunaan. Jika Anda tidak memiliki penggunaan Spot terbaru, kami memberi Anda batas default rendah yang selaras dengan batas permintaan Spot Anda.
- Batas konfigurasi permintaan — Kami dapat membatasi jumlah konfigurasi permintaan baru dalam jangka waktu 24 jam jika kami mendeteksi pola yang tidak terkait dengan tujuan penggunaan fitur skor penempatan Spot. Jika Anda mencapai batas, Anda dapat mencoba kembali konfigurasi permintaan yang telah Anda gunakan, tetapi Anda tidak dapat menentukan konfigurasi permintaan baru hingga periode 24 jam berikutnya.
- Jumlah minimum tipe instans — Jika Anda menentukan tipe instans, Anda harus menentukan setidaknya tiga tipe instans yang berbeda; jika tidak, Amazon EC2 akan menampilkan skor penempatan Spot rendah. Demikian pula, jika Anda menentukan atribut instans, atribut itu harus menyelesaikan setidaknya tiga tipe instans yang berbeda. Tipe instans dianggap berbeda jika mereka memiliki nama yang berbeda. Misalnya, m5.8xlarge, m5a.8xlarge, dan m5.12xlarge, semua dianggap berbeda.

Biaya

Tidak ada biaya tambahan karena menggunakan fitur skor penempatan Spot.

Cara kerja skor penempatan Spot

Saat menggunakan fitur skor penempatan Spot, pertama-tama tentukan kebutuhan komputasi untuk Instans Spot Anda, lalu Amazon EC2 akan menampilkan skor 10 Wilayah teratas atau Zona Ketersediaan tempat permintaan Spot Anda kemungkinan berhasil. Setiap Wilayah atau Zona Ketersediaan dinilai pada skala 1 hingga 10, dengan 10 menunjukkan bahwa permintaan Spot Anda sangat mungkin berhasil, dan 1 menunjukkan bahwa permintaan Spot Anda tidak mungkin berhasil. Hasil Zonal dapat mencakup Zona Ketersediaan dan Zona Lokal.

Untuk menggunakan fitur skor penempatan Spot, ikuti langkah-langkah berikut:

- Langkah 1: Tentukan kebutuhan Spot Anda
- Langkah 2: Filter respons skor penempatan Spot
- Langkah 3: Tinjau rekomendasi

• Langkah 4: Gunakan rekomendasi

Langkah 1: Tentukan kebutuhan Spot Anda

Pertama, tentukan kapasitas Spot target yang Anda inginkan dan kebutuhan komputasi Anda, sebagai berikut:

1. Tentukan kapasitas Spot target, dan unit kapasitas target opsional.

   Anda dapat menentukan kapasitas Spot target yang Anda inginkan dalam hal jumlah instans atau vCPU, atau dalam hal jumlah memori dalam MiB. Untuk menentukan kapasitas target dalam jumlah vCPU atau jumlah memori, Anda harus menentukan unit kapasitas target sebagai atau `vcpu` atau `memory-mib`. Jika tidak, default ditentukan ke jumlah instans.

   Dengan menentukan kapasitas target Anda dalam hal jumlah vCPU atau jumlah memori, Anda dapat menggunakan unit ini saat menghitung total kapasitas. Misalnya, jika Anda ingin menggunakan campuran instans dengan ukuran berbeda, Anda dapat menentukan kapasitas target sebagai jumlah total vCPU. Fitur skor penempatan Spot kemudian mempertimbangkan setiap tipe instans dalam permintaan berdasarkan jumlah vCPU-nya, dan menghitung jumlah total vCPU daripada jumlah total instans saat menjumlahkan kapasitas target.

   Misalnya, Anda menentukan total kapasitas target adalah 30 vCPU, dan daftar tipe instans Anda terdiri dari c5.xlarge (4 vCPU), m5.2xlarge (8 vCPU), dan r5.large (2 vCPU). Untuk mencapai total 30 vCPU, Anda bisa mendapatkan campuran 2 c5.xlarge (2*4 vCPU), 2 m5.2xlarge (2*8 vCPU), dan 3 r5.large (3*2 vCPU).
2. Tentukan tipe instans atau atribut instans.

   Anda dapat menentukan tipe instans yang akan digunakan, atau Anda dapat menentukan atribut instans yang Anda perlukan untuk kebutuhan komputasi, lalu biarkan Amazon EC2 mengidentifikasi tipe instans yang memiliki atribut tersebut. Pemilihan ini dikenal sebagai pemilihan tipe instans berbasis atribut.

   Anda tidak dapat menentukan tipe instans sekaligus atribut instans dalam permintaan skor penempatan Spot yang sama.

   Jika Anda menentukan tipe instans, Anda harus menentukan setidaknya tiga tipe instans yang berbeda, jika tidak, Amazon EC2 akan mengembalikan skor penempatan Spot rendah. Demikian pula, jika Anda menentukan atribut instans, atribut itu harus menyelesaikan setidaknya tiga tipe instans yang berbeda.

Untuk contoh berbagai cara menentukan kebutuhan Spot Anda, lihat Contoh konfigurasi.

Langkah 2: Filter respons skor penempatan Spot

Amazon EC2 menghitung skor penempatan Spot untuk setiap Wilayah atau Zona Ketersediaan, dan menampilkan 10 Wilayah teratas atau 10 Zona Ketersediaan teratas di mana permintaan Spot Anda kemungkinan akan berhasil. Defaultnya adalah menampilkan daftar Wilayah dengan skornya. Jika berencana untuk meluncurkan semua kapasitas Spot Anda ke dalam satu Zona Ketersediaan, lebih baik Anda meminta daftar Zona Ketersediaan dengan skornya.

Anda dapat menentukan filter Wilayah untuk mempersempit Wilayah yang akan ditampilkan dalam respons.

Anda dapat menggabungkan filter Wilayah dan permintaan Zona Ketersediaan dengan skornya. Dengan cara ini, Zona Ketersediaan dengan skornya dibatasi untuk Wilayah yang telah Anda filter. Untuk menemukan Zona Ketersediaan dengan skor tertinggi di suatu Wilayah, tentukan hanya Wilayah tersebut, dan responsnya akan menampilkan daftar skor dari semua Zona Ketersediaan di Wilayah tersebut.

Secara default, skor penempatan Spot mengabaikan Zona Lokal. Untuk memasukkannya, atur `IncludeLocalZones` ke`true`. Untuk skor regional, kapasitas Zona Lokal diperhitungkan dalam skor untuk Wilayah induknya. Untuk skor zonal, respons dapat mencakup skor Zona Lokal individu.

Langkah 3: Tinjau rekomendasi

Skor penempatan Spot untuk setiap Wilayah atau Zona Ketersediaan dihitung berdasarkan kapasitas target, komposisi jenis instans, tren penggunaan Spot saat ini, dan waktu permintaan. Karena kapasitas Spot terus berfluktuasi, permintaan skor penempatan Spot yang sama dapat menghasilkan skor yang berbeda ketika dihitung pada waktu yang berbeda.

Wilayah dan Zona Ketersediaan diberi skor pada skala 1 hingga 10. Skor 10 menunjukkan bahwa permintaan Spot Anda sangat mungkin—tetapi tidak dijamin—akan berhasil. Skor 1 menunjukkan bahwa permintaan Spot Anda tidak mungkin berhasil. Skor yang sama mungkin ditampilkan untuk Wilayah atau Zona Ketersediaan yang berbeda.

Jika skor rendah ditampilkan, Anda dapat mengedit kebutuhan komputasi Anda dan menghitung ulang skor. Anda juga dapat meminta rekomendasi skor penempatan Spot untuk kebutuhan komputasi yang sama pada waktu yang berbeda dalam sehari.

Langkah 4: Gunakan rekomendasi

Skor penempatan Spot hanya relevan jika permintaan Spot Anda memiliki konfigurasi yang persis sama dengan konfigurasi skor penempatan Spot (kapasitas target, unit kapasitas target, dan tipe instans atau atribut instans), dan dikonfigurasi untuk menggunakan strategi alokasi `capacity-optimized`. Jika tidak, kemungkinan mendapatkan kapasitas Spot yang tersedia tidak akan selaras dengan skor.

Jika skor Anda termasuk Zona Lokal, konfigurasikan permintaan Spot berikutnya untuk menggunakan Zona Lokal yang sama. Jika tidak, kemungkinan mendapatkan kapasitas mungkin tidak selaras dengan skor.

Meskipun skor penempatan Spot berfungsi sebagai pedoman, dan tidak ada skor yang menjamin bahwa permintaan Spot Anda akan terpenuhi sepenuhnya atau sebagian, Anda dapat menggunakan informasi berikut untuk mendapatkan hasil terbaik:

- Gunakan konfigurasi yang sama — Skor penempatan Spot hanya relevan jika konfigurasi permintaan Spot (kapasitas target, unit kapasitas target, dan tipe instans atau atribut instans) di grup Auto Scaling, Armada EC2, atau Armada Spot Anda sama dengan yang Anda masukkan untuk mendapatkan skor penempatan Spot.

  Jika Anda menggunakan pemilihan tipe instans berdasarkan atribut dalam permintaan skor penempatan Spot, Anda dapat menggunakan pemilihan tipe instans berdasarkan atribut untuk mengonfigurasi grup Auto Scaling, Armada EC2, atau Armada Spot. Untuk informasi selengkapnya, lihat Membuat grup instance campuran menggunakan pemilihan jenis instance berbasis atribut dan Tentukan atribut untuk pemilihan tipe instans untuk EC2 Fleet atau Spot Fleet.

  > **Note**
  >
  > Jika Anda menentukan kapasitas target berdasarkan jumlah vCPU atau jumlah memori, dan Anda menentukan tipe instans dalam konfigurasi skor penempatan Spot, perhatikan bahwa saat ini Anda tidak dapat membuat konfigurasi ini di grup Auto Scaling, Armada EC2, atau Armada Spot. Namun, Anda harus secara manual mengatur pembobotan instans dengan menggunakan parameter `WeightedCapacity`.

- Gunakan strategi alokasi **`capacity-optimized`** — Skor berapa pun mengasumsikan bahwa permintaan armada Anda akan dikonfigurasi untuk menggunakan semua Zona Ketersediaan (untuk meminta kapasitas di seluruh Wilayah) atau satu Zona Ketersediaan (jika meminta kapasitas dalam satu Zona Ketersediaan) dan strategi alokasi Spot `capacity-optimized` untuk

permintaan Anda agar kapasitas Spot berhasil. Jika Anda menggunakan strategi alokasi lain, seperti `lowest-price`, kemungkinan mendapatkan kapasitas Spot yang tersedia tidak akan selaras dengan skor.

- Segera bertindak berdasarkan skor — Rekomendasi skor penempatan Spot mencerminkan kapasitas Spot yang tersedia pada saat permintaan, dan konfigurasi yang sama dapat menghasilkan skor yang berbeda bila dihitung pada waktu yang berbeda karena fluktuasi kapasitas Spot. Meskipun skor 10 berarti permintaan kapasitas Spot Anda sangat mungkin—tetapi tidak dijamin—berhasil, untuk hasil terbaik kami sarankan Anda segera bertindak berdasarkan skor. Kami juga menyarankan Anda untuk mendapatkan skor baru setiap kali Anda mencoba permintaan kapasitas.

### Izin yang diperlukan untuk skor penempatan Spot

Secara default, identitas IAM (pengguna, peran, atau grup) tidak memiliki izin untuk digunakanthe section called “Skor penempatan Spot”. Untuk mengizinkan identitas IAM menggunakan skor penempatan Spot, Anda harus membuat kebijakan IAM yang memberikan izin untuk menggunakan tindakan API `ec2:GetSpotPlacementScores` EC2. Anda kemudian lampirkan kebijakan ke identitas IAM yang memerlukan izin ini.

Berikut ini adalah contoh kebijakan IAM yang memberikan izin untuk menggunakan tindakan EC2 API `ec2:GetSpotPlacementScores`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:GetSpotPlacementScores",
            "Resource": "*"
        }
    ]
}
```

Untuk informasi tentang pembuatan kebijakan IAM, lihat Mengedit kebijakan IAM di Panduan Pengguna IAM.

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Hitung skor penempatan Spot

Anda dapat menghitung skor penempatan Spot berdasarkan kapasitas target dan persyaratan komputasi. Untuk informasi selengkapnya, lihat the section called “Cara kerja skor penempatan Spot”.

Izin yang diperlukan

Pastikan Anda memiliki izin yang diperlukan. Untuk informasi selengkapnya, lihat the section called “Izin yang diperlukan”.

Opsi

- Hitung menggunakan atribut instance
- Hitung menggunakan tipe instance
- Hitung menggunakan AWS CLI

Mencari solusi otomatis?  Alih-alih mengikuti langkah-langkah manual dalam panduan pengguna ini, Anda dapat membuat dasbor pelacak skor penempatan Spot yang secara otomatis menangkap dan menyimpan skor di Amazon CloudWatch. Untuk informasi selengkapnya, lihat  Panduan Membangun Dasbor Pelacak Skor Penempatan Spot di AWS.

Zona Lokal

Untuk menyertakan Zona Lokal saat menggunakan konsol Amazon EC2, akun Anda harus mengaktifkan setidaknya satu Zona Lokal. Untuk informasi selengkapnya, lihat Mem ulai dengan AWS Zona Lok AWS al di Panduan Pengguna Zona Lokal.

Hitung menggunakan atribut instance

Untuk menghitung skor penempatan Spot dengan menentukan atribut instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih panah bawah di samping Minta Inst ans Spot dan pilih Hitung Skor Penempatan Spot.
4. Pilih Masukkan persyaratan.
5. Untuk Kapasitas target, masukkan kapasitas yang Anda inginkan dalam hal jumlah instans atau vCPU, atau jumlah memori (MiB).
6. Untuk Persyaratan tipe instans, untuk menentukan kebutuhan komputasi Anda dan agar Amazon EC2 dapat mengidentifikasi tipe instans yang optimal untuk kebutuhan ini, pilih Tentukan atribut instans yang sesuai dengan kebutuhan komputasi Anda.
7. Untuk vCPU, masukkan jumlah minimum dan maksimum vCPU yang diinginkan. Untuk menentukan tanpa batasan, pilih Tanpa minimum, Tanpa maksimum, atau keduanya.
8. Untuk Memori (GiB), masukkan jumlah memori minimum dan maksimum yang diinginkan. Untuk menentukan tanpa batasan, pilih Tanpa minimum, Tanpa maksimum, atau keduanya.
9. Untuk Arsitektur CPU, pilih arsitektur instans yang diperlukan.
10. (Opsional) Untuk Atribut instans tambahan, Anda dapat secara opsional menentukan satu atau lebih atribut untuk mengekspresikan kebutuhan komputasi Anda secara lebih mendetail. Setiap atribut tambahan menambahkan batasan lebih lanjut ke permintaan Anda. Anda dapat menghilangkan atribut tambahan; ketika dihilangkan, nilai default digunakan. Untuk deskripsi setiap atribut dan nilai defaultnya, lihat get-spot-placement-scores.
11. (Opsional) Untuk menampilkan tipe instans dengan atribut tertentu, perluas Pratinjau tipe instans yang cocok. Untuk mengecualikan tipe instans agar tidak digunakan dalam evaluasi penempatan Anda, pilih instans, lalu pilih Kecualikan tipe instans yang dipilih.
12. Pilih Muat skor penempatan, dan tinjau hasilnya.
13. (Opsional) Untuk menampilkan skor penempatan Spot untuk Wilayah tertentu, di Wilayah untuk dievaluasi, pilih Wilayah yang akan dievaluasi, lalu pilih Hitung skor penempatan.

14. (Opsional) Di bawah Opsi Skor, pilih Ser takan Zona Lokal.
15. (Opsional) Untuk menampilkan skor penempatan Spot untuk Zona Ketersediaan di Wil , pilih kotak centang Berikan skor penempatan per Zona Ketersediaan. Daftar Zona Ketersediaan yang dinilai akan bermanfaat jika Anda berencana untuk meluncurkan semua kapasitas Spot Anda ke dalam satu Zona Ketersediaan.
16. (Opsional) Untuk mengedit kebutuhan komputasi Anda dan mendapatkan skor penempatan baru, pilih Edit, buat penyesuaian yang diperlukan, lalu pilih Hitung skor penempatan.

Hitung menggunakan tipe instance

Untuk menghitung skor penempatan Spot dengan menentukan tipe instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih panah bawah di samping Minta Inst ans Spot dan pilih Hitung Skor Penempatan Spot.
4. Pilih Masukkan persyaratan.
5. Untuk Kapasitas target, masukkan kapasitas yang Anda inginkan dalam hal jumlah instans atau vCPU, atau jumlah memori (MiB).
6. Untuk Persyaratan tipe instans, untuk menentukan tipe instans yang akan digunakan, pilih Pilih tipe instans secara manual.
7. Pilih Pilih tipe instans, pilih tipe instans yang akan digunakan, lalu pilih Pilih. Untuk menemukan tipe instans dengan cepat, Anda dapat menggunakan bilah filter untuk memfilter tipe instans berdasarkan properti yang berbeda.
8. Pilih Muat skor penempatan, dan tinjau hasilnya.
9. (Opsional) Untuk menampilkan skor penempatan Spot untuk Wilayah tertentu, di Wilayah untuk dievaluasi, pilih Wilayah yang akan dievaluasi, lalu pilih Hitung skor penempatan.
10. (Opsional) Di bawah Opsi Skor, pilih Ser takan Zona Lokal.
11. (Opsional) Untuk menampilkan skor penempatan Spot untuk Zona Ketersediaan di Wil , pilih kotak centang Berikan skor penempatan per Zona Ketersediaan. Daftar Zona Ketersediaan yang dinilai akan bermanfaat jika Anda berencana untuk meluncurkan semua kapasitas Spot Anda ke dalam satu Zona Ketersediaan.
12. (Opsional) Untuk mengedit daftar tipe instans dan mendapatkan skor penempatan baru, pilih Edit, buat penyesuaian yang diperlukan, lalu pilih Hitung skor penempatan.

Hitung menggunakan AWS CLI

Hitung skor penempatan Spot

1. (Opsional) Untuk menghasilkan semua kemungkinan parameter yang dapat ditentukan untuk konfigurasi skor penempatan Spot, gunakan perintah get-spot-placement-score dan parameternya `--generate-cli-skeleton`.

```
aws ec2 get-spot-placement-scores \
    --region us-east-1 \
    --generate-cli-skeleton
```

Berikut ini adalah output contoh.

```
{
    "InstanceTypes": [
        ""
    ],
    "TargetCapacity": 0,
    "TargetCapacityUnitType": "vcpu",
    "SingleAvailabilityZone": true,
    "RegionNames": [
        ""
    ],
    "InstanceRequirementsWithMetadata": {
        "ArchitectureTypes": [
            "x86_64_mac"
        ],
        "VirtualizationTypes": [
            "hvm"
        ],
        "InstanceRequirements": {
            "VCpuCount": {
                "Min": 0,
                "Max": 0
            },
            "MemoryMiB": {
                "Min": 0,
                "Max": 0
            },
            "CpuManufacturers": [
                "amd"
```

```
            ],
            "MemoryGiBPerVCpu": {
                "Min": 0.0,
                "Max": 0.0
            },
            "ExcludedInstanceTypes": [
                ""
            ],
            "InstanceGenerations": [
                "previous"
            ],
            "SpotMaxPricePercentageOverLowestPrice": 0,
            "OnDemandMaxPricePercentageOverLowestPrice": 0,
            "BareMetal": "excluded",
            "BurstablePerformance": "excluded",
            "RequireHibernateSupport": true,
            "NetworkInterfaceCount": {
                "Min": 0,
                "Max": 0
            },
            "LocalStorage": "included",
            "LocalStorageTypes": [
                "hdd"
            ],
            "TotalLocalStorageGB": {
                "Min": 0.0,
                "Max": 0.0
            },
            "BaselineEbsBandwidthMbps": {
                "Min": 0,
                "Max": 0
            },
            "AcceleratorTypes": [
                "fpga"
            ],
            "AcceleratorCount": {
                "Min": 0,
                "Max": 0
            },
            "AcceleratorManufacturers": [
                "amd"
            ],
            "AcceleratorNames": [
                "vu9p"
```

```
            ],
            "AcceleratorTotalMemoryMiB": {
                "Min": 0,
                "Max": 0
            }
        }
    },
    "DryRun": true,
    "MaxResults": 0,
    "NextToken": "",
    "IncludeLocalZones": true
}
```

2. Buat file konfigurasi JSON menggunakan output dari langkah sebelumnya, dan konfigurasikan sebagai berikut:

   a. Untuk `TargetCapacity`, masukkan kapasitas Spot yang Anda inginkan dalam hal jumlah instans atau vCPU, atau jumlah memori (MiB).

   b. Untuk `TargetCapacityUnitType`, masukkan unit untuk kapasitas target. Jika Anda menghilangkan parameter ini, defaultnya adalah `units`.

      Nilai yang valid: `units` (yang diterjemahkan ke jumlah contoh) | `vcpu` | `memory-mib`

   c. Untuk `SingleAvailabilityZone`, tentukan `true` untuk respons yang menampilkan daftar Zona Ketersediaan dengan skornya. Daftar Zona Ketersediaan yang dinilai akan bermanfaat jika Anda berencana untuk meluncurkan semua kapasitas Spot Anda ke dalam satu Zona Ketersediaan. Jika Anda menghilangkan parameter ini, parameter defaultnya adalah `false`, dan respons akan menampilkan daftar Wilayah dengan skornya.

   d. (Opsional) Untuk`IncludeLocalZones`, tentukan `true` untuk menyertakan Zona Lokal. Untuk skor regional, kapasitas Zona Lokal diperhitungkan terhadap Wilayah induk. Untuk skor zonal, respons dapat mencakup skor Zona Lokal individu. Jika dihilangkan atau disetel ke`false`, Zona Lokal diabaikan.

   e. (Opsional) Untuk `RegionNames`, tentukan Wilayah yang akan digunakan sebagai filter. Anda harus menentukan kode Wilayah, misalnya, `us-east-1`.

      Dengan filter Wilayah, respons hanya menampilkan Wilayah yang Anda tentukan. Jika Anda menentukan `true` untuk `SingleAvailabilityZone`, respons hanya menampilkan Zona Ketersediaan di Wilayah yang ditentukan.

   f. Anda dapat memasukkan salah satu `InstanceTypes` atau `InstanceRequirements`, tetapi tidak keduanya dalam konfigurasi yang sama.

Tentukan salah satu dari berikut ini dalam konfigurasi JSON Anda:

- Untuk menentukan daftar tipe instans, tentukan tipe instans dalam parameter `InstanceTypes`. Tentukan setidaknya tiga tipe instans yang berbeda. Jika Anda hanya menentukan satu atau dua tipe instans, skor penempatan Spot menampilkan skor rendah. Untuk daftar tipe instans, lihat Tipe Instans Amazon EC2.
- Untuk menentukan atribut instans sehingga Amazon EC2 akan mengidentifikasi tipe instans yang cocok dengan atribut tersebut, tentukan atribut yang terletak di struktur `InstanceRequirements`.

  Anda harus memberikan nilai untuk `VCpuCount`, `MemoryMiB`, dan `CpuManufacturers`. Anda dapat menghilangkan atribut lainnya; saat dihilangkan, nilai default digunakan. Untuk deskripsi setiap atribut dan nilai defaultnya, lihat get-spot-placement-scores.

Untuk contoh konfigurasi, lihat Contoh konfigurasi.

3. Untuk mendapatkan skor penempatan Spot untuk persyaratan yang Anda tentukan dalam file JSON, gunakan perintah get-spot-placement-scores, dan tentukan nama serta jalur ke file JSON Anda dengan menggunakan parameter `--cli-input-json`.

```
aws ec2 get-spot-placement-scores \
    --region us-east-1 \
    --cli-input-json file://file_name.json
```

Contoh keluaran jika `SingleAvailabilityZone` disetel ke `false` atau dihilangkan (jika dihilangkan, defaultnya menjadi`false`) - daftar  skor Wilayah yang diberi skor

```
"SpotPlacementScores": [
    {
        "Region": "us-east-1",
        "Score": 7
    },
    {
        "Region": "us-west-1",
        "Score": 5
    },
   ...
```

Contoh keluaran jika `SingleAvailabilityZone` disetel ke `true` — daftar skor Zona Ketersediaan dan, bila `IncludeLocalZones` ada`true`, Zona Lokal dikembalikan.

```
"SpotPlacementScores": [
    {
        "Region": "us-east-1",
        "AvailabilityZoneId": "use1-az1",
        "Score": 8
    },
    {
        "Region": "us-east-1",
        "AvailabilityZoneId": "usw2-az3",
        "Score": 6
    },
   ...
```

Contoh konfigurasi

Saat menggunakan AWS CLI, Anda dapat menggunakan contoh konfigurasi berikut.

Contoh konfigurasi

- Contoh: Tentukan tipe instans dan kapasitas target
- Contoh: Tentukan tipe instans, dan kapasitas target dalam hal memori
- Contoh: Tentukan atribut untuk pemilihan tipe instans berdasarkan atribut
- Contoh: Tentukan atribut untuk pemilihan tipe instans berdasarkan atribut dan tampilkan daftar Zona Ketersediaan dengan skornya

Contoh: Tentukan tipe instans dan kapasitas target

Contoh konfigurasi berikut menentukan tiga tipe instans yang berbeda dan kapasitas Spot target adalah 500 Instans Spot.

```
{
    "InstanceTypes": [
        "m5.4xlarge",
        "r5.2xlarge",
        "m4.4xlarge"
    ],
```

```
    "TargetCapacity": 500
}
```

Contoh: Tentukan tipe instans, dan kapasitas target dalam hal memori

Contoh konfigurasi berikut menentukan tiga tipe instans yang berbeda dan kapasitas Spot target 500.000 MiB memori, di mana jumlah Instans Spot yang akan diluncurkan harus menyediakan total 500.000 MiB memori.

```
{
    "InstanceTypes": [
        "m5.4xlarge",
        "r5.2xlarge",
        "m4.4xlarge"
    ],
    "TargetCapacity": 500000,
    "TargetCapacityUnitType": "memory-mib"
}
```

Contoh: Tentukan atribut untuk pemilihan tipe instans berdasarkan atribut

Contoh konfigurasi berikut dikonfigurasi untuk pemilihan tipe instans berdasarkan atribut, dan diikuti dengan penjelasan teks tentang contoh konfigurasi.

```
{
    "TargetCapacity": 5000,
    "TargetCapacityUnitType": "vcpu",
    "InstanceRequirementsWithMetadata": {
        "ArchitectureTypes": ["arm64"],
        "VirtualizationTypes": ["hvm"],
        "InstanceRequirements": {
            "VCpuCount": {
                "Min": 1,
                "Max": 12
            },
            "MemoryMiB": {
                "Min": 512
            }
        }
    }
}
```

### `InstanceRequirementsWithMetadata`

Untuk menggunakan pemilihan instans berdasarkan atribut, Anda harus menyertakan struktur `InstanceRequirementsWithMetadata` dalam konfigurasi Anda, dan menentukan atribut yang diinginkan untuk Instans Spot.

Pada contoh sebelumnya, atribut instans yang diperlukan ditentukan berikut ini:

- `ArchitectureTypes` — Tipe arsitektur dari tipe instans harus `arm64`.
- `VirtualizationTypes` — Tipe virtualisasi dari tipe instans harus `hvm`.
- `VCpuCount` — Tipe instans harus memiliki minimal 1 dan maksimal 12 vCPU.
- `MemoryMiB` — Tipe instans harus memiliki memori minimal 512 MiB. Dengan menghilangkan parameter `Max`, Anda menunjukkan bahwa tidak ada batas maksimum.

Perhatikan bahwa ada beberapa atribut opsional lain yang dapat Anda tentukan. Untuk daftar atribut, lihat get-spot-placement-scores.

### `TargetCapacityUnitType`

Parameter `TargetCapacityUnitType` menentukan unit untuk kapasitas target. Dalam contoh, kapasitas targetnya adalah `5000` dan tipe unit kapasitas targetnya adalah `vcpu`, yang keduanya menentukan kapasitas target yang diinginkan sebesar 5.000 vCPU, di mana jumlah Instans Spot yang akan diluncurkan harus menyediakan total 5.000 vCPU.

Contoh: Tentukan atribut untuk pemilihan tipe instans berdasarkan atribut dan tampilkan daftar Zona Ketersediaan dengan skornya

Contoh konfigurasi berikut dikonfigurasi untuk pemilihan tipe instans berdasarkan atribut. Dengan menentukan `"SingleAvailabilityZone": true`, respons akan menampilkan daftar Zona Ketersediaan dengan skornya.

```
{
    "TargetCapacity": 1000,
    "TargetCapacityUnitType": "vcpu",
    "SingleAvailabilityZone": true,
    "InstanceRequirementsWithMetadata": {
        "ArchitectureTypes": ["arm64"],
        "VirtualizationTypes": ["hvm"],
        "InstanceRequirements": {
```

```
            "VCpuCount": {
                "Min": 1,
                "Max": 12
            },
            "MemoryMiB": {
                "Min": 512
            }
        }
    }
}
```

## Melacak biaya Instans Spot menggunakan umpan data Instans Spot

Untuk membantu Anda memahami biaya untuk Instans Spot Anda, Amazon EC2 menyediakan umpan data yang menjelaskan penggunaan dan harga Instans Spot Anda. Umpan data ini dikirim ke bucket Amazon S3 yang Anda tentukan saat Anda berlangganan umpan data.

File umpan data tiba di bucket Anda biasanya satu jam sekali. Jika Anda tidak menjalankan Instans Spot selama jam tertentu, Anda tidak menerima file data feed untuk jam itu.

Setiap jam penggunaan Instans Spot biasanya tercakup dalam satu file data. File-file ini dikompresi (gzip) sebelum dikirim ke bucket Anda. Amazon EC2 dapat menulis banyak file selama jam penggunaan tertentu di mana file berukuran besar (misalnya, ketika konten file untuk satu jam melebihi 50 MB sebelum kompresi).

> **Note**
>
> Anda hanya dapat membuat satu umpan data Instance Spot per Akun AWS.

Umpan data Instans Spot didukung di semua Wil AWS ayah kecuali China (Beijing), China (Ningxia), AWS GovCloud (AS), dan Wil ayah yang dinonaktifkan secara default.

Daftar Isi


- Nama dan format file umpan data
- Persyaratan bucket Amazon S3
- Berlangganan ke umpan data Instans Spot Anda
- Melihat data di umpan data Anda
- Hapus umpan data Instans Spot Anda

Nama dan format file umpan data

Nama file feed data Instans Spot menggunakan format berikut (dengan tanggal dan jam dalam UTC):

```
bucket-name.s3.amazonaws.com/optional-prefix/aws-account-id.YYYY-MM-DD-HH.n.unique-id.gz
```

Misalnya, jika nama bucket Anda adalah **amzn-s3-demo-bucket** dan prefiks Anda adalah **my-prefix**, nama file Anda mirip dengan yang berikut ini:

```
amzn-s3-demo-bucket.s3.amazonaws.com/my-prefix/111122223333.2023-12-09-07.001.b959dbc6.gz
```

Untuk informasi selengkapnya tentang nama bucket, lihat Aturan penamaan bucket di Panduan Pengguna Amazon S3.

File data feed instans Spot dibatasi tab. Setiap baris dalam file data sesuai dengan satu jam penggunaan instance dan berisi bidang yang tercantum dalam tabel berikut.

| Bidang | Deskripsi |
| --- | --- |
| `Timestamp` | Stempel waktu yang digunakan untuk menentukan harga yang dikenakan untuk penggunaan instans ini. |
| `UsageType` | Tipe penggunaan dan tipe instans yang dikenai biaya. Untuk Instans Spot, `m1.small` bidang ini diatur ke `SpotUsage` . Untuk semua tipe instans lainnya, bidang ini diatur ke `SpotUsage:` {instance-type}. Sebagai conto,.`SpotUsage:c1.medium` |
| `Operation` | Produk yang ditagihkan. Untuk Instans Spot Linux, bidang ini diatur ke `RunInstances` . Untuk Instans Spot Windows, bidang ini diatur ke `RunInstances:0002` . Penggunaan spot dikelompokkan menurut Zona Ketersediaan. |
| `InstanceID` | ID Instans Spot yang menghasilkan penggunaan instans ini. |

| Bidang | Deskripsi |
| --- | --- |
| `MyBidID` | ID untuk permintaan Instans Spot yang menghasilkan penggunaan instans ini. |
| `MyMaxPrice` | Harga maksimum yang ditentukan untuk permintaan Spot ini. |
| `MarketPrice` | Harga Spot pada waktu yang ditentukan di bidang `Timestamp` . |
| `Charge` | Harga yang dikenakan untuk penggunaan instans ini. |
| `Version` | Versi umpan data. Versi yang memungkinkan adalah versi 1.0. |

Persyaratan bucket Amazon S3

Saat Anda berlangganan umpan data, Anda harus menentukan bucket Amazon S3 untuk menyimpan file umpan data tersebut.

Sebelum Anda memilih bucket Amazon S3 untuk umpan data, pertimbangkan hal berikut:

- Anda harus memiliki izin `FULL_CONTROL` ke bucket. Jika Anda adalah pemilik bucket, Anda memiliki izin ini secara default. Jika tidak, pemilik bucket harus memberikan izin Akun AWS ini kepada Anda.
- Saat Anda berlangganan umpan data, izin ini digunakan untuk memperbarui bucket ACL untuk memberikan `FULL_CONTROL` izin akun umpan AWS data. Akun AWS umpan data menulis file umpan data ke bucket. Jika akun Anda tidak memiliki izin yang diperlukan, file data feed tidak dapat ditulis ke bucket. Untuk informasi selengkapnya, lihat Log yang dikirim ke Amazon S3 di Panduan CloudWatch Pengguna Amazon Log.

  Jika Anda memperbarui ACL dan menghapus izin untuk akun AWS umpan data, file umpan data tidak dapat ditulis ke bucket. Anda harus berlangganan kembali umpan data untuk menerima file data umpan.
- Setiap file umpan data memiliki ACL-nya sendiri (terpisah dari ACL untuk bucket). Pemilik bucket memiliki izin `FULL_CONTROL` ke file data. Akun AWS umpan data memiliki izin baca dan tulis.
- Jika Anda menghapus langganan umpan data, Amazon EC2 tidak menghapus izin baca dan tulis untuk akun umpan AWS data baik pada bucket maupun file data. Anda harus menghapus izin ini sendiri.

- Jika Anda mengenkripsi bucket Amazon S3 menggunakan enkripsi sisi server dengan AWS KMS kunci yang disimpan di AWS Key Management Service (SSE-KMS), Anda harus menggunakan kunci yang dikelola pelanggan. Untuk informasi selengkapnya, lihat enkripsi sisi server bucket Amazon S3 di Panduan Pengguna Amazon CloudWatch Logs.

Berlangganan ke umpan data Instans Spot Anda

Anda dapat berlangganan feed data Instance Spot kapan saja. Anda tidak dapat menyelesaikan tugas ini menggunakan konsol Amazon EC2.

Jika Anda mendapatkan kesalahan bahwa bucket tidak memiliki izin yang cukup, lihat artikel berikut untuk informasi pemecahan masalah: Memecahkan masalah umpan data untuk Instans Spot.

AWS CLI

Untuk berlangganan umpan data Anda

Gunakan perintah create-spot-datafeed-subscription.

Untuk menentukan bucket dengan awalan, gunakan contoh berikut:

```
aws ec2 create-spot-datafeed-subscription \
    --bucket amzn-s3-demo-bucket \
    --prefix my-prefix
```

Untuk menentukan bucket tanpa awalan, gunakan contoh berikut:

```
aws ec2 create-spot-datafeed-subscription \
    --bucket amzn-s3-demo-bucket
```

PowerShell

Untuk berlangganan umpan data Anda

Gunakan New-EC2SpotDatafeedSubscription cmdlet.

Untuk menentukan bucket dengan awalan, gunakan contoh berikut:

```
New-EC2SpotDatafeedSubscription `
    -Bucket amzn-s3-demo-bucket `
    -Prefix my-prefix
```

Untuk menentukan bucket tanpa awalan, gunakan contoh berikut:

```
New-EC2SpotDatafeedSubscription `
    -Bucket amzn-s3-demo-bucket
```

Melihat data di umpan data Anda

Di dalam Konsol Manajemen AWS, buka AWS CloudShell. Gunakan perintah s3 sync berikut untuk mendapatkan file.gz dari bucket S3 untuk umpan data Anda dan menyimpannya di folder yang Anda tentukan.

```
aws s3 sync s3://amzn-s3-demo-bucket ./data-feed
```

Untuk menampilkan isi file .gz, ubah ke folder tempat Anda menyimpan konten bucket S3.

```
cd data-feed
```

Gunakan perintah ls untuk melihat nama-nama file. Gunakan perintah zcat dengan nama file untuk menampilkan konten file terkompresi. Hal berikut menunjukkan contoh perintah.

```
zcat  111122223333.2023-12-09-07.001.b959dbc6.gz
```

Berikut ini adalah output contoh.

```
#Version: 1.0
#Fields: Timestamp UsageType Operation InstanceID MyBidID MyMaxPrice MarketPrice Charge
 Version
2023-12-09 07:13:47 UTC USE2-SpotUsage:c7a.medium       RunInstances:SV050
 i-0c3e0c0b046e050df     sir-pwq6nmfp    0.0510000000 USD        0.0142000000 USD
  0.0142000000 USD        1
```

Hapus umpan data Instans Spot Anda

Setelah selesai dengan umpan data Instance Spot, Anda dapat menghapusnya.

AWS CLI

Untuk menghapus umpan data Anda

Gunakan perintah delete-spot-datafeed-subscription.

```
aws ec2 delete-spot-datafeed-subscription
```

PowerShell

Untuk menghapus umpan data Anda

Gunakan Remove-EC2SpotDatafeedSubscription cmdlet.

```
Remove-EC2SpotDatafeedSubscription
```

## Service-linked peran untuk permintaan Instance Spot

Amazon EC2 menggunakan peran tertaut layanan untuk izin yang diperlukan untuk memanggil layanan AWS lain atas nama Anda. Peran terkait layanan adalah jenis peran IAM unik yang ditautkan langsung ke. Layanan AWS Service-linked peran menyediakan cara aman untuk mendelegasikan izin Layanan AWS karena hanya layanan tertaut yang dapat mengambil peran terkait layanan. Untuk informasi selengkapnya, lihat Service-linked peran di Panduan Pengguna IAM.

Amazon EC2 menggunakan peran terkait layanan bernamaAWSServiceRoleForEC2Spot untuk meluncurkan dan mengelola Instans Spot atas nama Anda.

Izin yang diberikan oleh AWSServiceRoleForEC2Spot

Penggunaan Amazon EC2 AWSServiceRoleForEC2Spot untuk menyelesaikan tindakan berikut:

- `ec2:DescribeInstances` – Menjelaskan Instans Spot
- `ec2:StopInstances` – Menghentikan Instans Spot
- `ec2:StartInstances` – Memulai Instans Spot

Membuat peran tertaut layanan

Dalam sebagian besar situasi, Anda tidak perlu membuat peran tertaut layanan secara manual. Amazon EC2 membuatAWSServiceRoleForEC2Spot peran terkait layanan saat pertama kali Anda meminta Instans Spot menggunakan konsol.

Jika Anda memiliki permintaan Instans Spot aktif sebelum Oktober 2017, ketika Amazon EC2 mulai mendukung peran terkait layanan ini, Amazon EC2 membuat peran tersebut

AWSServiceRoleForEC2Spot di akun Anda. AWS Untuk informasi selengkapnya, lihat Peran Baru Muncul di Akun Saya dalam Panduan Pengguna IAM.

Jika Anda menggunakan AWS CLI atau API untuk meminta Instans Spot, Anda harus terlebih dahulu memastikan bahwa peran ini ada.

Untuk membuat AWSServiceRoleForEC2Spot menggunakan konsol

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Pilih Buat peran.
4. Di halaman Pilih tipe entitas tepercaya, pilih EC2, EC2 - Instans Spot, Berikutnya: Izin.
5. Di halaman berikutnya, pilih Next:Review.
6. Di halaman Tinjau, pilih Buat peran.

Untuk membuat AWSServiceRoleForEC2Spot menggunakan AWS CLI

Gunakan perintah create-service-linked-role sebagai berikut.

```
aws iam create-service-linked-role --aws-service-name spot.amazonaws.com
```

Jika Anda tidak lagi perlu menggunakan Instans Spot, kami menyarankan Anda untuk menghapusAWSServiceRoleForEC2Spot wewenang. Setelah peran ini dihapus dari akun Anda, Amazon EC2 akan membuat peran lagi jika Anda meminta Instans Spot.

Memberikan akses ke kunci yang dikelola pelanggan untuk digunakan dengan AMI terenkripsi dan snapshot EBS

Jika Anda menentukan AMI terenkripsi atau snapshot Amazon EBS terenkripsi untuk Instans Spot Anda dan Anda menggunakan kunci yang dikelola pelanggan untuk enkripsi, Anda harus memberikan izin AWSServiceRoleForEC2Spot peran untuk menggunakan kunci yang dikelola pelanggan sehingga Amazon EC2 dapat meluncurkan Instans Spot atas nama Anda. Untuk melakukannya, Anda harus menambahkan pemberian izin ke kunci yang dikelola pelanggan, seperti yang ditunjukkan dalam prosedur berikut.

Ketika memberikan izin, pemberian izin merupakan alternatif dari kebijakan kunci. Untuk informasi selengkapnya, lihat Menggunakan Pemberian Izin dan Menggunakan Kebijakan Kunci di AWS KMS dalam Panduan Developer AWS Key Management Service .

Untuk memberikan AWSServiceRoleForEC2Spot izin peran untuk menggunakan kunci yang dikelola pelanggan

- Gunakan perintah create-grant untuk menambahkan hibah ke kunci yang dikelola pelanggan dan untuk menentukan prinsipal (peran AWSServiceRoleForEC2Spot terkait layanan) yang diberi izin untuk melakukan operasi yang diizinkan oleh hibah. Kunci yang dikelola pelanggan ditentukan oleh parameter `key-id` dan ARN kunci yang dikelola pelanggan. Kepala sekolah ditentukan oleh`grantee-principal` parameter dan ARN dariAWSServiceRoleForEC2Spot peran terkait layanan.

```
aws kms create-grant \
    --region us-east-1 \
    --key-id arn:aws:kms:us-
east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
spot.amazonaws.com/AWSServiceRoleForEC2Spot \
    --operations "Decrypt" "Encrypt" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
 "ReEncryptTo"
```

## Kuota Instans Spot

Ada kuota untuk jumlah Instans Spot yang berjalan dan permintaan Instans Spot yang tertunda per Akun AWS per Wilayah. Setelah permintaan Instans Spot yang tertunda dipenuhi, permintaan tidak lagi dihitung dalam kuota karena instans yang berjalan dihitung dalam kuota.

Kuota Instans Spot dikelola berdasarkan jumlah unit pemrosesan pusat virtual (vCPU) yang digunakan atau akan digunakan Instans Spot berjalan Anda sembari menunggu pemenuhan permintaan Instans Spot terbuka. Jika Anda menghentikan Instans Spot tetapi tidak membatalkan permintaan Instans Spot, permintaan tersebut diperhitungkan dalam kuota vCPU Instans Spot Anda hingga Amazon EC2 mendeteksi pengakhiran Instans Spot dan menutup permintaan.

Kami menyediakan jenis kuota berikut untuk Instans Spot.

| Nama | Default | Dapat disesuaikan |
|---|---|---|
| Semua Permintaan Instans Spot DL | 0 | Ya |
| Semua Permintaan Instans F Spot | 0 | Ya |

| Nama | Default | Dapat disesuaikan |
| --- | --- | --- |
| Semua Permintaan Instans Spot G dan VT | 0 | Ya |
| Semua Permintaan Instans Spot Inf | 0 | Ya |
| Semua Permintaan Instans Spot P | 0 | Ya |
| Semua Permintaan Instans Spot Standar (A, C, D, H, I, M, R, T, Z) | 5 | Ya |
| Semua Permintaan Instans Spot Trn | 0 | Ya |
| Semua Permintaan Instans Spot X | 0 | Ya |

Meskipun Amazon EC2 secara otomatis menyesuaikan kuota Instans Spot berdasarkan penggunaan Anda, Anda dapat meminta peningkatan kuota jika perlu. Misalnya, jika Anda ingin meluncurkan lebih banyak Instans Spot daripada yang diizinkan kuota Anda saat ini, Anda dapat meminta peningkatan kuota. Anda juga dapat meminta peningkatan kuota jika Anda mengirimkan permintaan Instans Spot dan Anda menerima kesalahan `Max spot instance count exceeded`. Untuk meminta kenaikan kuota, gunakan konsol Kuota Layanan yang dijelaskan di Kuota layanan Amazon EC2.

Anda dapat meluncurkan kombinasi tipe instans apa pun yang memenuhi kebutuhan aplikasi Anda yang berubah. Misalnya, dengan kuota Semua Permintaan Instans Spot Standar sejumlah 256 vCPU, Anda dapat meminta 32 Instans Spot `m5.2xlarge` (32 x 8 vCPU) atau 16 Instans Spot `c5.4xlarge` (16 x 16 vCPU).

Dengan integrasi CloudWatch metrik Amazon, Anda dapat memantau penggunaan EC2 terhadap kuota Anda. Anda juga dapat mengonfigurasi alarm untuk memperingatkan saat sudah mendekati kuota. Untuk informasi selengkapnya, lihat Kuota Layanan dan CloudWatch alarm Amazon di Panduan Pengguna Kuota Layanan Mem Pengguna Amazon.

## Host Khusus Amazon EC

Amazon EC2 Dedicated Host adalah server fisik yang sepenuhnya didedikasikan untuk Anda gunakan. Secara opsional, Anda dapat memilih untuk berbagi kapasitas instans dengan AWS akun lain. Untuk informasi selengkapnya, lihat Cross-account Berbagi Host Khusus Amazon EC2.

Host Khusus memberikan visibilitas dan kontrol atas penempatan instans dan mendukung afinitas host. Ini berarti Anda dapat meluncurkan dan menjalankan instance pada host tertentu, dan Anda dapat memastikan bahwa instance hanya berjalan pada host tertentu. Untuk informasi selengkapnya, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.

Host Khusus menyediakan dukungan Bring Your Own License (BYOL) yang komprehensif. Mereka memungkinkan Anda untuk menggunakan lisensi perangkat lunak per-socket, per-core, atau per-VM yang ada, termasuk Windows Server, SQL Server, SUSE Linux Enterprise Server, Red Hat Enterprise Linux, atau lisensi perangkat lunak lain yang terikat pada VM, soket, atau inti fisik, tunduk pada ketentuan lisensi Anda.

Jika Anda mengharuskan instans berjalan pada perangkat keras khusus, tetapi Anda tidak memerlukan visibilitas atau kontrol atas penempatan instans, dan Anda tidak perlu menggunakan lisensi perangkat lunak per soket atau per-inti, Anda dapat mempertimbangkan untuk menggunakan Instans Khusus sebagai gantinya. Instans Khusus dan Host Khusus keduanya dapat digunakan untuk meluncurkan instans Amazon EC2 ke server fisik khusus. Tidak ada perbedaan performa, keamanan, atau fisik di antara Instans Khusus dan instans pada Host Khusus. Namun, ada beberapa perbedaan utama di antara mereka. Tabel berikut menyoroti beberapa perbedaan utama antara Instans Khusus dan Host Khusus:

| | Host Khusus | Instans Khusus |
|---|---|---|
| Server fisik khusus | Server fisik dengan kapasitas instance yang sepenuhnya didedikasikan untuk penggunaan Anda. | Server fisik yang didedikasikan untuk satu akun pelanggan. |
| Berbagi kapasitas instans | Dapat berbagi kapasitas instance dengan akun lain. | Tidak didukung |
| Penagihan | Per-host penagihan | Per-instance penagihan |
| Visibilitas soket, inti, dan ID host | Memberikan visibilitas dalam jumlah soket dan inti fisik | Tidak ada visibilitas |
| Afinitas host dan instans | Memungkinkan Anda melakukan deployment instans Anda secara | Tidak didukung |

| | Host Khusus | Instans Khusus |
|---|---|---|
| | konsisten ke server fisik yang sama seiring waktu | |
| Penempatan instans tertarget | Memberikan visibilitas dan kontrol tambahan atas cara penempatan instans di server fisik | Tidak didukung |
| Pemulihan instans otomatis | Didukung. Untuk informasi selengkapnya, lihat Pemulihan Host Khusus Amazon EC2. | Didukung |
| Bawa Lisensi Sendiri (BYOL) | Didukung | Dukungan parsial* |
| Reservasi Kapasitas | Tidak didukung | Didukung |

* Microsoft SQL Server dengan License Mobility melalui Jaminan Perangkat Lunak, sedangkan lisensi Windows Virtual Desktop Access (VDA) dapat digunakan dengan Instans Khusus.

Untuk informasi selengkapnya tentang metadata instans, lihat Instans Khusus Amazon EC2.

## Larangan Host Khusus

Sebelum Anda mengalokasikan Host Khusus, perhatikan batasan dan larangan berikut:

- Untuk menjalankan RHEL dan SUSE Linux di Host Khusus, Anda harus membawa AMI Anda sendiri. Anda tidak dapat menggunakan AMI Linux RHEL dan SUSE yang ditawarkan oleh AWS atau yang tersedia AWS Marketplace dengan Host Khusus. Untuk informasi lebih lanjut tentang cara membuat AMI Anda sendiri, lihatBawa lisensi perangkat lunak Anda sendiri ke Amazon EC2 Dedicated Host.

  Pembatasan ini tidak berlaku untuk host yang dialokasikan untuk instans memori tinggi (`u-6tb1.metal`, `u-9tb1.metal`, `u-12tb1.metal`, `u-18tb1.metal`, dan `u-24tb1.metal`). AMI Linux RHEL dan SUSE yang ditawarkan oleh AWS atau yang tersedia AWS Marketplace dapat digunakan dengan host ini.

- Ada batasan jumlah menjalankan Host Khusus per keluarga instans per akun AWS per Wilayah. Kuota hanya berlaku untuk menjalankan instans. Jika instans Anda tertunda, berhenti, atau dihentikan, instans tersebut tidak akan dihitung ke dalam kuota Anda. Untuk melihat kuota akun Anda, atau meminta peningkatan kuota, gunakan konsol Kuota Layanan.
- Grup Auto Scaling didukung saat menggunakan templat peluncuran yang menentukan grup sumber daya host. Untuk informasi selengkapnya, lihat Membuat template peluncuran menggunakan pengaturan lanjutan di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
- Instans Amazon RDS tidak didukung.
- Tingkat Penggunaan AWS Gratis tidak tersedia untuk Host Khusus.
- Kontrol penempatan instans mengacu pada pengelolaan peluncuran instans ke Host Khusus. Anda tidak dapat meluncurkan Host Khusus ke grup penempatan.
- Jika Anda mengalokasikan host untuk tipe instans tervirtualisasi, Anda tidak dapat mengubah tipe instans menjadi tipe `.metal` instans setelah host dialokasikan. Misalnya, jika Anda mengalokasikan host untuk tipe instans `m5.large`, Anda tidak dapat mengubah tipe instans menjadi `m5.metal`.

  Demikian pula, jika Anda mengalokasikan host untuk tipe `.metal` instans, Anda tidak dapat memodifikasi tipe instans menjadi tipe instans virtual setelah host dialokasikan. Misalnya, jika Anda mengalokasikan host untuk tipe instans `m5.metal`, Anda tidak dapat mengubah tipe instans menjadi `m5.large`.

Daftar Isi


- Harga dan penagihan Host Khusus Amazon EC2
- Konfigurasi kapasitas instans Host Khusus Amazon EC2
- Instans T3 yang dapat meledak di Host Khusus Amazon EC2
- Bawa lisensi perangkat lunak Anda sendiri ke Amazon EC2 Dedicated Host
- Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host
- Mengalokasikan Host Khusus Amazon EC2 untuk digunakan di akun Anda
- Luncurkan instans Amazon EC2 di Host Khusus Amazon EC2
- Luncurkan instans Amazon EC2 ke grup sumber daya host
- Ubah pengaturan penempatan otomatis untuk Host Khusus Amazon EC2 yang ada
- Ubah jenis instans yang didukung untuk Host Khusus Amazon EC2 yang ada
- Ubah penyewaan dan afinitas Host Khusus Amazon EC2 untuk instans Amazon EC2

- Merilis Host Khusus Amazon EC2
- Bermigrasi ke Nitro-based Host Khusus Amazon EC2
- Beli Reservasi Host Khusus untuk diskon penagihan Host Khusus
- Cross-account Berbagi Host Khusus Amazon EC2
- Host Khusus Amazon EC2 aktif AWS Outposts
- Pemulihan Host Khusus Amazon EC2
- Pemeliharaan host untuk Host Khusus Amazon EC2
- Pantau status Host Khusus Amazon EC2 Anda
- Lacak perubahan konfigurasi Host Khusus Amazon EC2 menggunakan AWS Config

## Harga dan penagihan Host Khusus Amazon EC2

Harga untuk Host Khusus bervariasi menurut opsi pembayaran.

Opsi pembayaran

- On-Demand Tuan Rumah Khusus
- Reservasi Host Khusus
- Paket Tabungan
- Harga untuk Windows Server pada Host Khusus

On-Demand Tuan Rumah Khusus

On-Demand penagihan diaktifkan secara otomatis saat Anda mengalokasikan Host Khusus ke akun Anda.

On-Demand Harga untuk Host Khusus bervariasi menurut keluarga instans dan Wilayah. Anda membayar per detik (dengan minimal 60 detik) untuk Host Khusus yang aktif, terlepas dari jumlah atau ukuran instans yang Anda pilih untuk diluncurkan. Untuk informasi selengkapnya tentang On-Demand harga, lihat Harga Host Khusus Amazon EC2. On-Demand

Anda dapat melepaskan Host On-Demand Khusus kapan saja untuk berhenti membebankan biaya untuk itu. Untuk informasi tentang pelepasan Host Khusus, lihat Merilis Host Khusus Amazon EC2.

### Reservasi Host Khusus

Reservasi Host Khusus memberikan diskon penagihan dibandingkan dengan menjalankan Host On-Demand Khusus. Reservasi tersedia dalam tiga opsi pembayaran:

- Tanpa Uang Muka—Reservasi Tanpa Uang Muka memberi Anda diskon untuk penggunaan Host Khusus selama jangka waktu tertentu dan tidak memerlukan pembayaran di muka. Tersedia dalam jangka waktu satu tahun dan tiga tahun. Hanya beberapa keluarga instans yang mendukung jangka waktu tiga tahun untuk Reservasi Tanpa Uang Muka.
- Sebagian Di Muka—Sebagian dari reservasi harus dibayar di muka dan sisa jam dalam jangka waktu tersebut ditagih dengan tarif yang didiskon. Tersedia dalam jangka waktu satu tahun dan tiga tahun.
- Lunas di Muka—Memberikan harga efektif terendah. Tersedia dalam jangka waktu satu tahun dan tiga tahun serta mencakup seluruh biaya selama jangka waktu itu di muka, tanpa biaya tambahan di masa mendatang.

Anda harus memiliki Host Khusus yang aktif di akun Anda sebelum dapat membeli reservasi. Setiap reservasi dapat mencakup satu host atau lebih yang mendukung keluarga instans yang sama dalam satu Zona Ketersediaan. Reservasi diterapkan ke keluarga instans di host, bukan ukuran instans. Jika Anda memiliki tiga Host Khusus dengan ukuran instans berbeda (`m4.xlarge`, `m4.medium`, dan `m4.large` ) Anda dapat mengaitkan satu reservasi `m4` dengan semua Host Khusus tersebut. Keluarga instans dan Zona Ketersediaan reservasi harus cocok dengan Host Khusus yang ingin Anda kaitkan dengannya.

Saat reservasi dikaitkan dengan Host Khusus, Host Khusus tidak dapat dilepaskan hingga jangka waktu reservasi berakhir.

Untuk informasi selengkapnya tentang harga reservasi, lihat Harga Host Khusus Amazon EC2.

### Paket Tabungan

Paket Tabungan adalah model penetapan harga fleksibel yang menawarkan penghematan signifikan dibandingkan On-Demand Instans. Dengan Savings Plans, Anda membuat komitmen dengan jumlah penggunaan yang konsisten, dalam USD per jam, selama jangka waktu satu atau tiga tahun. Ini memberi Anda fleksibilitas untuk menggunakan Host Khusus yang paling sesuai dengan kebutuhan Anda dan terus menghemat uang, daripada membuat komitmen untuk Host Khusus tertentu. Untuk informasi selengkapnya, lihat Panduan Pengguna AWS Savings Plans.

> **Note**
>
> Savings Plans tidak didukung dengan Host Khusus `u-6tb1.metal`, `u-9tb1.metal`, `u-12tb1.metal`, `u-18tb1.metal`, dan `u-24tb1.metal`.

Harga untuk Windows Server pada Host Khusus

Tunduk pada persyaratan lisensi Microsoft, Anda dapat membawa lisensi Windows Server dan SQL Server yang ada ke Host Khusus. Tidak ada biaya tambahan untuk penggunaan perangkat lunak jika Anda memilih untuk membawa lisensi Anda sendiri.

Selain itu, Anda juga dapat menggunakan AMI Windows Server disediakan oleh Amazon untuk menjalankan Windows Server versi terbaru pada Host Khusus. Ini umum untuk skenario di mana Anda memiliki lisensi SQL Server yang memenuhi syarat untuk dijalankan di Host Khusus, tetapi memerlukan Windows Server untuk menjalankan beban kerja SQL Server. AMI Windows Server yang disediakan oleh Amazon hanya didukung pada jenis instans generasi saat ini.  Untuk informasi selengkapnya, lihat  Harga Host Khusus Amazon EC2.

## Konfigurasi kapasitas instans Host Khusus Amazon EC2

Host Khusus mendukung berbagai konfigurasi (inti fisik, soket, dan vCPU) yang memungkinkan Anda menjalankan instans dari berbagai keluarga dan ukuran.

Saat mengalokasikan Host Khusus di akun, Anda dapat memilih konfigurasi yang mendukung baik satu tipe instans maupun beberapa tipe instans dalam keluarga instans yang sama. Jumlah instans yang dapat Anda jalankan di host tergantung pada konfigurasi yang Anda pilih.

Daftar Isi


- Dukungan tipe instans tunggal
- Dukungan tipe banyak instans


Dukungan tipe instans tunggal

Anda dapat mengalokasikan Host Khusus yang hanya mendukung satu tipe instans. Dengan konfigurasi ini, setiap instans yang Anda luncurkan di Host Khusus harus memiliki tipe instans yang sama, yang Anda tentukan saat mengalokasikan host.

Misalnya, Anda dapat mengalokasikan host yang hanya mendukung tipe `m5.4xlarge` instans. Dalam hal ini, Anda hanya dapat menjalankan `m5.4xlarge` instans di host tersebut.

Jumlah instans yang dapat Anda luncurkan ke host bergantung pada jumlah inti fisik yang disediakan oleh host, dan jumlah inti yang dikonsumsi oleh tipe instans yang ditentukan. Misalnya, jika Anda mengalokasikan host untuk `m5.4xlarge` instans, host menyediakan 48 core fisik, dan setiap `m5.4xlarge` instans mengkonsumsi 8 core fisik. Ini berarti Anda dapat meluncurkan hingga 6 instans pada host tersebut (48 core fisik/8 core per instans = 6 instans).

Dukungan tipe banyak instans

Anda dapat mengalokasikan Host Khusus yang mendukung banyak tipe instans dalam keluarga instans yang sama. Ini memungkinkan Anda menjalankan tipe instans yang berbeda pada host yang sama, selama mereka berada dalam keluarga instans yang sama dan host memiliki kapasitas instans yang memadai.

Misalnya, Anda dapat mengalokasikan host yang mendukung berbagai tipe instans dalam keluarga R5 instans. Dalam hal ini, Anda dapat meluncurkan kombinasi jenis R5 instans tertentu, seperti `r5.larger5.xlarge`,`r5.2xlarge`,, dan`r5.4xlarge`, pada host tersebut, dalam kapasitas inti fisik host.

Keluarga instans berikut mendukung Host Khusus dengan dukungan beberapa tipe instans:

- Tujuan umum: A1 | M5 | M5n | M6i | M7i | T3
- Komputasi dioptimalkan: C5 | C5n | C6i | C7i
- Memori dioptimalkan: R5 | R5b | R5n | R6i | R7i | R7iz | X2ieDN

Jumlah instans yang dapat Anda jalankan di host bergantung pada jumlah core fisik yang disediakan oleh host, dan jumlah core yang dikonsumsi oleh setiap tipe instans yang Anda jalankan di host. Misalnya, jika Anda mengalokasikan R5 host, yang menyediakan 48 core fisik, dan Anda menjalankan dua `r5.2xlarge` instans (4 core x 2 *`r5.4xlarge`* instance) dan tiga instance (8 core x 3 instance), instance tersebut menggunakan total 32 core, dan Anda mungkin dapat menjalankan kombinasi R5 instance tertentu selama mereka berada dalam 16 core yang tersisa.

Namun, untuk setiap keluarga instans, ada batasan jumlah instance yang dapat dijalankan untuk setiap jenis instans. Misalnya, Host R5 Khusus mendukung maksimal 2 `r5.8xlarge` instans, yang menggunakan 32 inti fisik. Dalam hal ini, R5 contoh tambahan dari tipe yang lebih kecil kemudian dapat digunakan untuk mengisi host ke kapasitas inti. Untuk jumlah jenis instans yang didukung untuk setiap keluarga instans, lihat Tabel Konfigur asi Host Khusus.

Tabel berikut menunjukkan contoh kombinasi tipe instans:

| Keluarga instans | Contoh kombinasi tipe instance |
| --- | --- |
| R5 | • Contoh 1: 4 x r5.4xlarge + 4 x r5.2xlarge<br>• Contoh 2: 1 x r5.12xlarge + 1 x r5.4xlarge + 1 x r5.2xlarge + 5 x r5.xlarge + 2 x r5.large |
| C5 | • Contoh 1: 1 x c5.9xlarge + 2 x c5.4xlarge + 1 x c5.xlarge<br>• Contoh 2: 4 x c5.4xlarge + 1 x c5.xlarge + 2 x c5.large |
| M5 | • Contoh 1: 4 x m5.4xlarge + 4 x m5.2xlarge<br>• Contoh 2: 1 x m5.12xlarge + 1 x m5.4xlarge + 1 x m5.2xlarge + 5 x m5.xlarge + 2 x m5.large |

Pertimbangan-pertimbangan

Ingatlah hal-hal berikut ini saat bekerja dengan Host Khusus yang mendukung banyak tipe instans:

- Menggunakan beberapa jenis instance pada host yang sama hanya dimungkinkan dalam keluarga instance yang sama.
- Saat mencampur jenis instans, untuk memaksimalkan pemanfaatan host, sebaiknya luncurkan tipe instans yang lebih besar terlebih dahulu diikuti dengan tipe instans yang lebih kecil.
  - Bergantung pada kombinasi dan urutan peluncuran jenis instans pada Host Khusus, mungkin secara fisik tidak memungkinkan untuk memaksimalkan pemanfaatan host. Saat mencampur jenis instance pada host, beberapa kapasitas mungkin tersedia di host tetapi tidak dapat digunakan. Misalnya, Anda mungkin melihat 16 vCPU tersedia di host r5n tetapi mungkin tidak dapat meluncurkan instance 4xlarge pada host meskipun r5n.4xlarge berjalan pada 16 vCPU.

> **Note**
>
> Jika Anda mengaktifkan Host Khusus A1 untuk beberapa jenis instans, Anda hanya dapat meluncurkan campuran `a1.xlarge` dan `a1.2xlarge` instance pada host tersebut. Jika Anda meluncurkan `a1.large` instance `a1.medium` atau pada host itu, Anda akan dibatasi untuk meluncurkan hanya lebih banyak dari jenis instans yang sama di host. Satu `a1.4xlarge` instance menghabiskan semua kapasitas pada host. Jika Anda memerlukan host untuk salah satu `a1.medium` `a1.large` instans, sebaiknya alokasikan host terpisah untuk jenis instans tersebut.

## Instans T3 yang dapat meledak di Host Khusus Amazon EC2

Host Khusus mendukung instans T3 performa dapat melonjak. Instans T3 menyediakan cara hemat biaya untuk menggunakan perangkat lunak lisensi BYOL Anda yang memenuhi syarat pada perangkat keras khusus. Jejak vCPU yang lebih kecil dari instans T3 memungkinkan Anda untuk mengonsolidasikan beban kerja Anda pada host yang lebih sedikit dan memaksimalkan pemanfaatan lisensi per inti Anda.

Host Khusus T3 paling cocok untuk menjalankan perangkat lunak BYOL dengan pemanfaatan CPU rendah hingga sedang. Beban kerja tersebut termasuk lisensi perangkat lunak per soket, per inti, atau per VM yang memenuhi syarat, seperti Windows Server, Windows Desktop, SQL Server, SUSE Enterprise Linux Server, Red Hat Enterprise Linux, dan Oracle Database. Contoh beban kerja yang cocok untuk Host Khusus T3 adalah basis data kecil dan menengah, desktop virtual, lingkungan pengembangan dan pengujian, repositori kode, dan prototipe produk. Host Khusus T3 tidak direkomendasikan untuk beban kerja dengan pemanfaatan CPU tinggi yang berkelanjutan atau untuk beban kerja yang mengalami lonjakan CPU yang berkorelasi secara bersamaan.

Instans T3 pada Host Khusus menggunakan model kredit yang sama dengan instans T3 pada perangkat keras penghunian bersama. Namun, mereka hanya mendukung mode kredit `standard`; mereka tidak mendukung mode kredit `unlimited`. Dalam mode `standard`, instans T3 di Host Khusus memperoleh, menggunakan, dan mengakumulasikan kredit dengan cara yang sama seperti instans yang dapat melonjak pada perangkat keras penghunian bersama. Mereka memberikan performa CPU dasar dengan kemampuan untuk melonjak di atas level acuan. Untuk melonjak di atas batas dasar, instans menggunakan kredit yang telah diakumulasikan dalam saldo kredit CPU. Ketika kredit akumulasi habis, pemanfaatan CPU diturunkan ke tingkat acuan. Untuk informasi selengkapnya tentang mode `standard`, lihat Cara kerja instans performa yang dapat melonjak standar.

Host Khusus T3 mendukung semua fitur yang ditawarkan oleh Host Khusus Amazon EC2, termasuk banyak ukuran instans pada satu host, grup sumber daya Host, dan BYOL.

Ukuran dan konfigurasi instans T3 yang didukung

Host Khusus T3 menjalankan instans T3 tujuan umum yang dapat melonjak, yang berbagi sumber daya CPU host dengan menyediakan performa CPU dasar dan kemampuan untuk melonjak ke tingkat yang lebih tinggi bila diperlukan. Hal ini memungkinkan Host Khusus T3, yang memiliki 48 inti, untuk mendukung hingga maksimum 192 instans per host. Untuk menggunakan sumber daya host secara efisien dan memberikan kinerja instans terbaik, algoritma penempatan instans Amazon EC2 secara otomatis menghitung jumlah instans yang didukung dan kombinasi ukuran instans yang dapat diluncurkan di host.

Host Khusus T3 mendukung beberapa tipe instans pada host yang sama. Semua ukuran instans T3 didukung pada Host Khusus. Anda dapat menjalankan berbagai kombinasi instans T3 hingga batas CPU host.

Tabel berikut mencantumkan tipe instans yang didukung, merangkum kinerja setiap tipe instans, dan menunjukkan jumlah maksimum instans dari setiap ukuran yang dapat diluncurkan.

| Tipe insta | vCPU | Memori (GiB) | Pemanfaatan CPU acuan per vCPU | Bandwidth lonjakan jaringan (Gbps) | Bandwidth lonjakan Amazon EBS (Mbps) | Jumlah maksimum instans per Host Khusus |
|---|---|---|---|---|---|---|
| t3.na | 2 | 0.5 | 5% | 5 | Hingga 2.085 | 192 |
| t3.m | 2 | 1 | 10% | 5 | Hingga 2.085 | 192 |
| t3.sn | 2 | 2 | 20% | 5 | Hingga 2.085 | 192 |
| t3.m | 2 | 4 | 20% | 5 | Hingga 2.085 | 192 |
| t3.la | 2 | 8 | 30% | 5 | 2.780 | 96 |
| t3.xl | 4 | 16 | 40% | 5 | 2.780 | 48 |
| t3.2x e | 8 | 32 | 40% | 5 | 2.780 | 24 |

Pantau pemanfaatan CPU untuk Host Khusus T3

Anda dapat menggunakan CloudWatch metrik `DedicatedHostCPUUtilization` Amazon untuk memantau pemanfaatan vCPU dari Host Khusus. Metrik tersedia di namespace `EC2` dan dimensi `Per-Host-Metrics`. Untuk informasi selengkapnya, lihat Metrik Host Khusus.

## Bawa lisensi perangkat lunak Anda sendiri ke Amazon EC2 Dedicated Host

Host Khusus memungkinkan Anda menggunakan lisensi perangkat lunak per soket, per inti, atau per VM yang ada. Saat Anda membawa lisensi Anda sendiri, Anda bertanggung jawab untuk mengelola lisensi Anda sendiri. Namun, Amazon EC2 memiliki fitur yang membantu Anda menjaga kepatuhan lisensi, seperti afinitas instans dan penempatan tertarget.

Ini adalah langkah-langkah umum yang harus diikuti untuk membawa gambar mesin berlisensi volume Anda sendiri ke Amazon EC2.

1. Verifikasi bahwa persyaratan lisensi yang mengontrol penggunaan gambar mesin Anda mengizinkan penggunaan dalam lingkungan cloud tervirtualisasi. Untuk informasi selengkapnya tentang Lisensi Microsoft, lihat Amazon Web Services dan Microsoft Licensing.
2. Setelah Anda memverifikasi bahwa image mesin Anda dapat digunakan dalam Amazon EC2, impor menggunakan VM Import/Export. Untuk informasi tentang cara mengimpor image mesin Anda, lihat Panduan Import/Export Pengguna VM.
3. Setelah Anda mengimpor gambar mesin, Anda dapat meluncurkan instans darinya ke Host Khusus yang aktif di akun Anda.
4. Saat Anda menjalankan instans ini, bergantung pada sistem operasi, Anda mungkin diminta untuk mengaktifkan instans ini di server KMS Anda sendiri (misalnya, Windows Server atau Windows SQL Server). Anda tidak dapat mengaktifkan AMI Windows yang diimpor pada server Amazon Windows KMS.

> **Note**
>
> Untuk melacak bagaimana gambar Anda digunakan AWS, aktifkan perekaman host di AWS Config. Anda dapat menggunakan AWS Config untuk merekam perubahan konfigurasi ke Host Khusus dan menggunakan output sebagai sumber data untuk pelaporan lisensi. Untuk informasi selengkapnya, lihat Lacak perubahan konfigurasi Host Khusus Amazon EC2 menggunakan AWS Config.

## Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host

Kontrol penempatan untuk Host Khusus terjadi pada level instans dan level host.

Auto-placement

Auto-placement dikonfigurasi pada tingkat host. Ini memungkinkan Anda untuk mengelola apakah instans yang Anda luncurkan diluncurkan ke host tertentu, atau ke host mana pun yang tersedia yang memiliki konfigurasi yang cocok.

Ketika penempatan otomatis dinonaktifkan untuk Host Khusus, itu hanya menerima peluncuran instance penyewaan host yang menentukan ID host uniknya. Ini adalah pengaturan default untuk Host Khusus baru.

Ketika penempatan otomatis di aktifkan untuk Host Khusus, ia menerima peluncuran instans penyewaan host yang tidak ditargetkan yang sesuai dengan konfigurasi jenis instansnya.

Saat meluncurkan sebuah instans, Anda perlu mengonfigurasi penghuniannya. Meluncurkan sebuah instans ke Host Khusus tanpa memberikan `HostId` yang spesifik memungkinkannya untuk diluncurkan pada Host Khusus yang memiliki penempatan otomatis yang diaktifkan dan yang cocok dengan tipe instansnya.

Afinitas host

Afinitas host dikonfigurasi pada tingkat instans. Ini menetapkan hubungan peluncuran antara sebuah instans dan Host Khusus.

Saat afinitas ditetapkan ke `Host`, sebuah instans yang diluncurkan ke host tertentu selalu dimulai ulang di host yang sama jika dihentikan. Ini berlaku untuk peluncuran tertarget dan tidak tertarget.

Saat afinitas diatur ke `Default`, dan Anda menghentikan serta memulai ulang instans, instans ini dapat dimulai ulang di semua host yang tersedia. Namun, ia mencoba untuk meluncurkan kembali ke Host Khusus terakhir yang dijalankannya (dengan upaya terbaik).

## Mengalokasikan Host Khusus Amazon EC2 untuk digunakan di akun Anda

Untuk mulai menggunakan Host Khusus, Anda harus mengalokasikannya terlebih dahulu di akun Anda. Setelah Anda mengalokasikan Host Khusus, kapasitas Host Khusus akan segera tersedia di akun Anda dan Anda dapat mulai meluncurkan instans ke Host Khusus.

Saat mengalokasikan Host Khusus di akun, Anda dapat memilih konfigurasi yang mendukung baik satu tipe instans maupun beberapa tipe instans dalam keluarga instans yang sama. Jumlah instans

yang dapat Anda jalankan di host tergantung pada konfigurasi yang Anda pilih. Untuk informasi selengkapnya, lihat Konfigurasi kapasitas instans Host Khusus Amazon EC2.

Console

Untuk mengalokasikan Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus lalu pilih Alokasi Host Khusus.
3. Di keluarga instans, pilih keluarga instans untuk Host Khusus.
4. Tentukan apakah Host Khusus mendukung banyak ukuran instans dalam keluarga instans yang dipilih, atau hanya tipe instans tertentu. Lakukan salah satu dari berikut ini.
   - Untuk mengonfigurasi Host Khusus agar mendukung banyak tipe instans dalam keluarga instans yang dipilih, pada Dukung beberapa tipe instans, pilih Aktifkan. Dengan mengaktifkannya, Anda akan dapat meluncurkan ukuran instans yang berbeda dari keluarga instans yang sama ke Host Khusus. Misalnya, jika Anda memilih keluarga instans `m5` dan memilih opsi ini, Anda dapat meluncurkan instans `m5.xlarge` dan `m5.4xlarge` ke Host Khusus.
   - Untuk mengonfigurasi Host Khusus agar mendukung satu tipe instans dalam keluarga instans yang dipilih, hapus Dukung beberapa tipe instans, lalu untuk Tipe instans, pilih tipe instans yang akan didukung. Dengan demikian, Anda akan dapat meluncurkan satu tipe instans pada Host Khusus. Misalnya, jika Anda memilih opsi ini dan menentukan `m5.4xlarge` sebagai tipe instans yang didukung, Anda hanya dapat meluncurkan instans `m5.4xlarge` ke Host Khusus.
5. Untuk Zona Ketersediaan, pilih Zona Ketersediaan untuk mengalokasikan Host Khusus.
6. Agar Host Khusus dapat menerima peluncuran instans tidak tertarget yang cocok dengan tipe instansnya, di Penempatan otomatis instans, pilih Aktifkan. Untuk informasi selengkapnya tentang penempatan otomatis, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.
7. Untuk mengaktifkan pemulihan host untuk Host Khusus, pada Pemulihan host, pilih Aktifkan. Untuk informasi selengkapnya, lihat Pemulihan Host Khusus Amazon EC2.
8. Untuk Kuantitas, masukkan jumlah Host Khusus yang akan dialokasikan.
9. (Opsional) Pilih Tambahkan tanda baru dan masukkan kunci tanda dan nilai tanda.
10. Pilih Alokasikan.

AWS CLI

Untuk mengalokasikan Host Khusus

Gunakan perintah allocate-hosts. Contoh berikut mengalokasikan Host Khusus yang mendukung beberapa jenis instans dari keluarga inst m5 ans di Avail us-east-1a ability Zone. Ini juga memungkinkan pemulihan host dan menonaktifkan penempatan otomatis.

```
aws ec2 allocate-hosts \
    --instance-family "m5" \
    --availability-zone "us-east-1a" \
    --auto-placement "off" \
    --host-recovery "on" \
    --quantity 1
```

Contoh berikut mengalokasikan Host Khusus yang mendukung peluncuran instans yang tidak ditargetkan di Zona Ketersediaan yang ditentukan, mengaktifkan pemulihan host, dan mengaktifkan penempatan otomatis.

```
aws ec2 allocate-hosts \
    --instance-type "m5.large" \
    --availability-zone "eu-west-1a" \
    --auto-placement "on" \
    --host-recovery "on" \
    --quantity 1
```

PowerShell

Untuk mengalokasikan Host Khusus

Gunakan New-EC2Host cmdlet. Contoh berikut mengalokasikan Host Khusus yang mendukung beberapa jenis instans dari keluarga inst m5 ans di Avail us-east-1a ability Zone. Host juga mengaktifkan pemulihan host dan penempatan otomatis dinonaktifkan.

```
New-EC2Host `
    -InstanceFamily m5 `
    -AvailabilityZone us-east-1a `
    -AutoPlacement Off `
    -HostRecovery On `
    -Quantity 1
```

Contoh berikut mengalokasikan Host Khusus yang mendukung peluncuran instans yang tidak ditargetkan di Zona Ketersediaan yang ditentukan dan memungkinkan pemulihan host.

```
New-EC2Host `
    -InstanceType m5.large `
    -AvailabilityZone eu-west-1a `
    -AutoPlacement On `
    -HostRecovery On `
    -Quantity 1
```

## Luncurkan instans Amazon EC2 di Host Khusus Amazon EC2

Setelah Anda mengalokasikan Host Khusus, Anda dapat meluncurkan instans ke dalamnya. Anda tidak dapat meluncurkan instans dengan penghunian `host` jika Anda tidak memiliki Host Khusus aktif dengan kapasitas ketersediaan yang cukup untuk tipe instans yang Anda luncurkan.

Pertimbangan-pertimbangan

- AMI SQL Server, SUSE, dan RHEL yang disediakan oleh Amazon EC2 tidak dapat digunakan dengan Host Khusus.
- Untuk Host Khusus yang mendukung banyak ukuran instans, kami menyarankan Anda untuk meluncurkan ukuran instans yang lebih besar terlebih dahulu, kemudian mengisi kapasitas instans yang tersisa dengan ukuran instans yang lebih kecil, sesuai kebutuhan.
- Sebelum Anda meluncurkan instans Anda, perhatikan batasannya. Untuk informasi selengkapnya, lihat Larangan Host Khusus.

Console

Untuk meluncurkan sebuah instans ke Host Khusus tertentu dari halaman Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Host Khusus di panel navigasi.
3. Di halaman Host Khusus, pilih host dan pilih Tindakan, Luncurkan Instans ke host.
4. Di bagian Gambar Aplikasi dan OS, pilih AMI dari daftar.
5. Di bagian tipe instans, pilih tipe instans yang akan diluncurkan.

> **Note**
>
> Jika Host Khusus mendukung satu tipe instans saja, tipe instans yang didukung akan dipilih secara default dan tidak dapat diubah.
> Jika Host Khusus mendukung banyak tipe instans, Anda harus memilih tipe instans dalam keluarga instans yang didukung berdasarkan kapasitas instans yang tersedia pada Host Khusus. Kami menyarankan Anda untuk meluncurkan ukuran instans yang lebih besar terlebih dahulu, kemudian mengisi kapasitas instans yang tersisa dengan ukuran instans yang lebih kecil, sesuai kebutuhan.

6. Di bagian Key pair, pilih key pair untuk diasosiasikan dengan instans.
7. Di bagian Detail lanjutan, untuk Affinity Tenancy, pilih salah satu dari yang berikut ini:

   - Non aktif — Afinitas host dinonaktifkan. Instans diluncurkan ke host yang ditentukan, tetapi tidak dijamin akan restart pada Host Khusus yang sama jika dihentikan.
   - ID Host Khusus — Afinitas host diaktifkan. Jika dihentikan, instance selalu restart pada host yang ditentukan ini jika memiliki kapasitas. Jika host tidak memiliki kapasitas, instance tidak dapat dimulai ulang; Anda harus membangun afinitas dengan host yang berbeda.

   Untuk informasi selengkapnya tentang Afinitas, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.

   > **Note**
   >
   > Opsi Penghunian dan Host telah dikonfigurasi sebelumnya berdasarkan host yang Anda pilih.

8. Konfigurasikan opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat Referensi untuk parameter konfigurasi EC2 instans Amazon.
9. Pilih Luncurkan instans.

Untuk meluncurkan sebuah instans ke Host Khusus menggunakan Wizard Peluncuran Instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, Luncurkan instans.

3. Di bagian Gambar Aplikasi dan OS, pilih AMI dari daftar.
4. Di bagian tipe instans, pilih tipe instans yang akan diluncurkan.
5. Di bagian Key pair, pilih key pair untuk diasosiasikan dengan instans.
6. Di bagian Detail lanjutan, lakukan hal berikut:

   a. Untuk Penghunian, pilih Host Khusus.

   b. Untuk Target host berdasarkan, pilih ID Host.

   c. Untuk ID host Target, pilih host yang akan meluncurkan instans.

   d. Untuk Affinity  Tenancy, pilih salah satu dari yang berikut ini:

      - Non  aktif — Afinitas host dinonaktifkan. Instans diluncurkan ke host yang ditentukan, tetapi tidak dijamin akan restart pada Host Khusus yang sama jika dihentikan.
      - ID Host Khusus — Afinitas host diaktifkan. Jika dihentikan, instance selalu restart pada host yang ditentukan ini jika memiliki kapasitas. Jika host tidak memiliki kapasitas, instance tidak dapat dimulai ulang; Anda harus membangun afinitas dengan host yang berbeda.

   Untuk informasi selengkapnya tentang Afinitas, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.
7. Konfigurasikan opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat Referensi untuk parameter konfigurasi EC2 instans Amazon.
8. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan sebuah instans ke Host Khusus

Gunakan perintah run-instance dan tentukan afinitas instance, tenancy, dan host dalam opsi. `--placement`

Untuk meluncurkan ke Host Khusus tertentu dengan afinitas host (instance selalu restart pada host yang sama jika dihentikan):

```
--placement Affinity=host,Tenancy=host,HostId=h-07879acf49EXAMPLE
```

Untuk meluncurkan ke Host Khusus tertentu tanpa afinitas host (instance dapat memulai ulang pada host yang tersedia):

```
--placement Tenancy=host,HostId=h-07879acf49EXAMPLE
```

Untuk meluncurkan ke Host Khusus yang tersedia dengan penempatan otomatis diaktifkan dan jenis instans yang cocok:

```
--placement Tenancy=host
```

PowerShell

Untuk meluncurkan sebuah instans ke Host Khusus

Gunakan New-EC2Instance cmdlet dan tentukan afinitas instance, tenancy, dan host dalam parameter. `-Placement`

Untuk meluncurkan ke Host Khusus tertentu dengan afinitas host (instance selalu restart pada host yang sama jika dihentikan):

```
-Placement_Affinity host `
-Placement_Tenancy host `
-Placement_HostId h-07879acf49EXAMPLE
```

Untuk meluncurkan ke Host Khusus tertentu tanpa afinitas host (instance dapat memulai ulang pada host yang tersedia):

```
-Placement_Tenancy host `
-Placement_HostId h-07879acf49EXAMPLE
```

Untuk meluncurkan ke Host Khusus yang tersedia dengan penempatan otomatis diaktifkan dan jenis instans yang cocok:

```
-Placement_Tenancy host
```

## Luncurkan instans Amazon EC2 ke grup sumber daya host

Host Khusus juga terintegrasi dengan AWS License Manager. Dengan License Manager, Anda dapat membuat grup sumber daya host, yang merupakan kumpulan Host Khusus yang dikelola sebagai

satu entitas. Saat membuat grup sumber daya host, Anda menentukan preferensi pengelolaan host, seperti alokasi otomatis dan lepas otomatis, untuk Host Khusus. Ini memungkinkan Anda meluncurkan instans ke Host Khusus tanpa mengalokasikan dan mengelola host tersebut secara manual. Untuk informasi selengkapnya, lihat Grup Sumber Daya Host di Panduan Pengguna AWS License Manager .

Saat Anda meluncurkan instans ke dalam grup sumber daya host yang memiliki Host Khusus dengan kapasitas instans yang tersedia, Amazon EC2 meluncurkan instans ke host tersebut. Jika grup sumber daya host tidak memiliki host dengan kapasitas instans yang tersedia, Amazon EC2 secara otomatis mengalokasikan host baru di grup sumber daya host, lalu meluncurkan instans tersebut ke host tersebut. Untuk informasi selengkapnya, lihat Grup Sumber Daya Host di Panduan Pengguna AWS License Manager .

Persyaratan dan batasan

- Membuat konfigurasi lisensi (lisensi yang dikelola sendiri) untuk grup sumber daya host bersifat opsional:
  - Jika Anda memerlukan konfigurasi lisensi saat membuat grup sumber daya host, Anda dapat meluncurkan instans hanya dari AMI dengan konfigurasi lisensi berbasis inti atau soket yang cocok.
  - Jika Anda membuat grup sumber daya host agar tidak memerlukan konfigurasi lisensi, Anda tidak perlu menyiapkan lisensi yang dikelola sendiri atau mengaitkan AMI.
- Anda tidak dapat menggunakan AMI SQL SQL Server, SUSE, atau RHEL yang disediakan oleh Amazon EC2 dengan Host Khusus.
- Anda tidak dapat menargetkan host tertentu dengan memilih ID host, dan Anda tidak dapat mengaktifkan afinitas instans saat meluncurkan sebuah instans ke dalam grup sumber daya host.

Console

Untuk meluncurkan sebuah instans ke dalam grup sumber daya host

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, Luncurkan instans.
3. Di bagian Gambar Aplikasi dan OS, pilih AMI dari daftar.
4. Di bagian tipe instans, pilih tipe instans yang akan diluncurkan.
5. Di bagian Key pair, pilih key pair untuk diasosiasikan dengan instans.

6. Di bagian Detail lanjutan, lakukan hal berikut:

   a. Untuk Penghunian, pilih Host Khusus.

   b. Untuk Host target oleh, pilih Grup sumber daya host.

   c. Untuk Grup sumber daya host penghunian, pilih grup sumber daya host di mana instans akan diluncurkan.

   d. Untuk Afinitas penghunian, lakukan salah satu hal berikut ini:

      - Pilih Nonaktif — Instans diluncurkan ke host yang ditentukan, tetapi tidak ada jaminan bahwa instans akan dimulai ulang pada Host Khusus yang sama jika dihentikan.
      - Pilih ID Host Khusus — Jika dihentikan, instans selalu dimulai ulang di host spesifik ini.

      Untuk informasi selengkapnya tentang Afinitas, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.

7. Konfigurasikan opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat Referensi untuk parameter konfigurasi EC2 instans Amazon.

8. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan sebuah instans ke dalam grup sumber daya host

Gunakan perintah run-instans. Dalam `--placement` opsi, hilangkan sewa dan tentukan ARN dari grup sumber daya host.

```
--placement HostResourceGroupArn=arn:aws:resource-groups:us-
east-2:123456789012:group/my-resource-group
```

PowerShell

Untuk meluncurkan sebuah instans ke dalam grup sumber daya host

Gunakan New-EC2Instance cmdlet. Dalam `-Placement` parameter, hilangkan sewa dan tentukan ARN dari grup sumber daya host.

```
-Placement_HostResourceGroupArn arn:aws:resource-groups:us-
east-2:123456789012:group/my-resource-group
```

## Ubah pengaturan penempatan otomatis untuk Host Khusus Amazon EC2 yang ada

Anda dapat mengubah pengaturan penempatan otomatis Host Khusus setelah mengalokasikannya ke AWS akun Anda.

### Console

Untuk memodifikasi penempatan otomatis Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih host dan pilih Tindakan, Ubah host.
4. Untuk Penempatan otomatis instans, pilih Aktifkan untuk mengaktifkan penempatan otomatis, atau kosongkan Aktifkan untuk menonaktifkan penempatan otomatis. Untuk informasi selengkapnya, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.
5. Pilih Simpan.

### AWS CLI

Untuk memodifikasi penempatan otomatis Host Khusus

Gunakan perintah modify-hosts.

```
aws ec2 modify-hosts \
    --auto-placement on \
    --host-ids h-012a3456b7890cdef
```

### PowerShell

Untuk memodifikasi penempatan otomatis Host Khusus

Gunakan Edit-EC2Host cmdlet.

```
Edit-EC2Host `
    -AutoPlacement 1 `
    -HostId h-012a3456b7890cdef
```

## Ubah jenis instans yang didukung untuk Host Khusus Amazon EC2 yang ada

Anda dapat memodifikasi Host Khusus untuk mengubah tipe instans yang didukungnya. Jika saat ini mendukung satu tipe instans, Anda dapat memodifikasinya untuk mendukung beberapa tipe instans dalam keluarga instans itu. Demikian pula, jika saat ini mendukung beberapa tipe instans, Anda dapat memodifikasinya untuk mendukung tipe instans tertentu saja.

Untuk mengubah Host Khusus agar mendukung banyak tipe instans, Anda harus terlebih dahulu menghentikan semua instans yang berjalan di host. Modifikasi membutuhkan waktu sekitar 10 menit untuk selesai. Transisi Host Khusus ke status `pending` saat modifikasi sedang berlangsung. Anda tidak dapat memulai instans yang berhenti atau meluncurkan instans baru pada Host Khusus saat berada di status `pending`.

Untuk mengubah Host Khusus yang mendukung banyak tipe instans agar hanya mendukung satu tipe instans, host tidak boleh memiliki instans yang sedang berjalan, atau instans yang sedang berjalan harus dari tipe instans yang Anda inginkan agar didukung oleh host. Misalnya, untuk mengubah host yang mendukung beberapa tipe instans di keluarga instans `m5` untuk mendukung instans `m5.large` saja, Host Khusus tidak boleh memiliki instans yang berjalan, atau hanya boleh memiliki instans `m5.large` yang berjalan di atasnya.

Jika Anda mengalokasikan host untuk tipe instans tervirtualisasi, Anda tidak dapat mengubah tipe instans menjadi tipe `.metal` instans setelah host dialokasikan. Misalnya, jika Anda mengalokasikan host untuk tipe instans `m5.large`, Anda tidak dapat mengubah tipe instans menjadi `m5.metal`. Demikian pula, jika Anda mengalokasikan host untuk tipe `.metal` instans, Anda tidak dapat memodifikasi tipe instans menjadi tipe instans virtual setelah host dialokasikan. Misalnya, jika Anda mengalokasikan host untuk tipe instans `m5.metal`, Anda tidak dapat mengubah tipe instans menjadi `m5.large`.

Console

Untuk mengubah tipe instans yang didukung untuk Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel Navigasi, pilih Host Khusus.
3. Pilih Host Khusus untuk memodifikasi dan pilih Tindakan, Ubah host.
4. Lakukan salah satu hal berikut, tergantung pada konfigurasi Host Khusus saat ini:

   - Jika Host Khusus saat ini mendukung tipe instans tertentu, Dukung beberapa tipe instans tidak diaktifkan, dan Tipe instans mencantumkan tipe instans yang didukung. Untuk

mengubah host agar mendukung banyak tipe dalam keluarga instans saat ini, pada Dukungan beberapa tipe instans, pilih Aktifkan.

Anda harus terlebih dahulu menghentikan semua instans yang berjalan pada host sebelum memodifikasinya untuk mendukung banyak tipe instans.

- Jika Host Khusus saat ini mendukung beberapa tipe instans dalam sebuah keluarga instans, Diaktifkan dipilih untuk Mendukung beberapa tipe instans. Untuk memodifikasi host agar mendukung tipe instans tertentu, pada Dukungan beberapa tipe instans, hapus Aktifkan, lalu pada Tipe instans, pilih tipe instans tertentu yang akan didukung.

  Anda tidak dapat mengubah keluarga instans yang didukung oleh Host Khusus.

5. Pilih Simpan.

AWS CLI

Untuk mengubah tipe instans yang didukung untuk Host Khusus

Gunakan perintah modify-hosts.

Contoh berikut memodifikasi Host Khusus untuk mendukung beberapa jenis instans dalam keluarga `m5` instans.

```
aws ec2 modify-hosts \
    --instance-family m5 \
    --host-ids h-012a3456b7890cdef
```

Contoh berikut memodifikasi Host Khusus untuk mendukung `m5.xlarge` instans saja.

```
aws ec2 modify-hosts \
    --instance-type m5.xlarge \
    --instance-family --host-ids h-012a3456b7890cdef
```

PowerShell

Untuk mengubah tipe instans yang didukung untuk Host Khusus

Gunakan Edit-EC2Host cmdlet.

Contoh berikut memodifikasi Host Khusus untuk mendukung beberapa jenis instans dalam keluarga `m5` instans.

```
Edit-EC2Host `
    -InstanceFamily m5 `
    -HostId h-012a3456b7890cdef
```

Contoh berikut memodifikasi Host Khusus untuk mendukung `m5.xlarge` instans saja.

```
Edit-EC2Host `
    -InstanceType m5.xlarge `
    -HostId h-012a3456b7890cdef
```

## Ubah penyewaan dan afinitas Host Khusus Amazon EC2 untuk instans Amazon EC2

Anda dapat mengubah penghunian instans setelah Anda meluncurkannya. Anda juga dapat mengubah afinitas instans Anda untuk menargetkan host tertentu atau mengizinkannya diluncurkan pada host khusus apa pun yang tersedia dengan atribut yang cocok di akun Anda. Untuk mengubah penghunian atau afinitas instans, instans tersebut harus ada dalam status `stopped`.

Detail sistem operasi instans Anda, dan apakah SQL Server diinstal, memengaruhi konversi apa yang didukung. Untuk informasi selengkapnya tentang jalur konversi penghunian yang tersedia untuk instans Anda, lihat Konversi penghunia di Panduan Pengguna Manajer Lisensi.

> **Note**
>
> Untuk instans T3, Anda harus meluncurkan instans pada Host Khusus untuk menggunakan penyewaan `host`. Untuk instans T3, Anda tidak dapat mengubah penghunian dari `host` ke `dedicated` atau `default`. Percobaan mengubah salah satu penghunian yang tidak didukung ini dapat mengakibatkan kode kesalahan `InvalidRequest`.

### Console

Untuk memodifikasi penghunian atau afinitas instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Instans dan pilih instans yang akan dimodifikasi.
3. Pilih Status instans, Berhenti.
4. Dengan instans yang dipilih, pilih Tindakan, Pengaturan instans, Modifikasi penempatan instans.

5. Pada halaman Mo difikasi penempatan instans, konfigurasikan yang berikut ini:

   - Penghunian—Pilih salah satu dari berikut:
     - Jalankan instans perangkat keras khusus — Meluncurkan instans sebagai Instans Khusus. Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2.
     - Meluncurkan instans pada Host Khusus — Meluncurkan instans ke Host Khusus dengan afinitas yang dapat dikonfigurasi.
   - Afinitas—Pilih salah satu dari berikut:
     - Instans ini dapat berjalan di salah satu host saya—Instans ini diluncurkan ke Host Khusus mana pun yang tersedia di akun Anda yang mendukung tipe instansnya.
     - Instans ini hanya dapat berjalan di host yang dipilih—Instans ini hanya dapat berjalan di Host Khusus yang dipilih untuk Host Target.
   - Target Host—Pilih Host Khusus tempat instans harus dijalankan. Jika tidak ada host target yang terdaftar, Anda mungkin tidak memiliki Host Khusus yang tersedia dan kompatibel di akun Anda.

   Untuk informasi selengkapnya, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.

6. Pilih Simpan.

AWS CLI

Untuk memodifikasi penghunian atau afinitas instans

Gunakan perintah modify- instance-placement. Contoh berikut mengubah afinitas instance yang ditentukan dari `default` untuk, `host`dan menentukan Host Khusus yang terkait dengan instans.

```
aws ec2 modify-instance-placement \
    --instance-id i-1234567890abcdef0 \
    --affinity host \
    --tenancy host \
    --host-id h-012a3456b7890cdef
```

PowerShell

Untuk memodifikasi penghunian atau afinitas instans

Gunakan Edit-EC2InstancePlacement cmdlet. Contoh berikut mengubah afinitas instans yang ditetapkan dari `default` menjadi `host`, dan menetapkan Host Khusus yang mempunyai afinitas dengan instans.

```
Edit-EC2InstancePlacement `
    -InstanceId i-1234567890abcdef0 `
    -Affinity host `
    -Tenancy host `
    -HostId h-012a3456b7890cdef
```

## Merilis Host Khusus Amazon EC2

Jika Anda tidak lagi memerlukan Host Khusus, Anda dapat menghentikan instans yang berjalan di host, mengarahkannya untuk diluncurkan di host yang berbeda, lalu melepaskan host.

Setiap instans yang berjalan di Host Khusus harus dihentikan sebelum Anda dapat merilis host. Instans ini dapat dimigrasikan ke Host Khusus lainnya di akun Anda sehingga Anda dapat terus menggunakannya. Langkah-langkah ini hanya berlaku untuk On-Demand Host Khusus.

Console

Untuk merilis Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Di halaman Host Khusus, pilih Host Khusus yang akan dirilis.
4. Pilih Tindakan, Rilis host.
5. Untuk mengonfirmasi, pilih Lepaskan.

AWS CLI

Untuk merilis Host Khusus

Gunakan perintah release-hosts.

```
aws ec2 release-hosts --host-ids h-012a3456b7890cdef
```

PowerShell

Untuk merilis Host Khusus

Gunakan Remove-EC2Host cmdlet.

```
Remove-EC2Host -HostId h-012a3456b7890cdef
```

Setelah merilis Host Khusus, Anda tidak dapat menggunakan kembali host atau ID host yang sama, dan Anda tidak lagi dikenakan tarif On-Demand penagihan untuk itu. Status Host Khusus diubah menjadi `released`, dan Anda tidak dapat meluncurkan instans apa pun ke host itu.

> **Note**
>
> Jika Anda baru saja melepas Host Khusus, mungkin perlu beberapa saat bagi host tersebut untuk tidak diperhitungkan dalam batas Anda. Selama waktu ini, Anda mungkin mengalami kesalahan `LimitExceeded` saat mencoba mengalokasikan Host Khusus baru. Jika ini masalahnya, coba alokasikan host baru lagi setelah beberapa menit.

Instans yang dihentikan masih tersedia untuk digunakan dan terdaftar di halaman Instans. Pengaturan penghunian `host` dipertahankan.

## Bermigrasi ke Nitro-based Host Khusus Amazon EC2

Sistem Nitro adalah kumpulan komponen perangkat keras dan perangkat lunak yang dibangun oleh AWS yang memungkinkan kinerja tinggi, ketersediaan tinggi, dan keamanan tinggi. Nitro-based Host Khusus menawarkan kinerja harga yang lebih baik dibandingkan dengan Xen-based Host Khusus. Jika Anda memiliki Host Xen-based Khusus di akun Anda, kami sarankan Anda memigrasikan beban kerja Anda ke Host Nitro-based Khusus. Untuk informasi selengkapnya, lihat AWS Nitro System.

Untuk bermigrasi dari Host Xen-based Khusus ke Host Nitro-based Khusus, Anda perlu memigrasikan Xen-based instans di Host Khusus ke jenis Nitro-based instans, mengalokasikan Host Nitro-based Khusus baru, lalu memindahkan Nitro-based instans yang dimigrasi ke Host Nitro-based Khusus baru.

Topik ini memberikan langkah-langkah terperinci untuk bermigrasi dari Host Xen-based Khusus ke Host Nitro-based Khusus.

Langkah migrasi

- Langkah 1: Identifikasi Host Xen-based Khusus Anda
- Langkah 2: Memigr Xen-based asikan Nitro-based instance ke jenis instans
- Langkah 3: Alokasikan Host Nitro-based Khusus
- Langkah 4: Pindahkan instance yang dimigrasi ke Host Nitro-based Khusus baru
- Langkah 5: Lepaskan Host Xen-based Khusus yang tidak digunakan

### Langkah 1: Identifikasi Host Xen-based Khusus Anda

Host Khusus berikut memenuhi syarat untuk dimigrasikan ke Host Nitro-based Khusus. Xen-based

- Tujuan umum: M3 | M4
- Komputasi dioptimalkan: C3 | C4
- Memori dioptimalkan: R3 | R4 | X1 | X1e
- Penyimpanan dioptimalkan: D2 | H1 | I2 | I3
- Komputasi yang dipercepat: G3 | P3

Untuk memeriksa apakah Anda memiliki Host Xen-based Khusus di akun Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Di bidang Pencarian, gunakan filter keluarga Instans untuk mencari Host Xen-based Khusus di atas. Misalnya, keluarga Instance = m3.

### Langkah 2: Memigr Xen-based asikan Nitro-based instance ke jenis instans

Instans yang berjalan di Xen-based Host Khusus juga Xen-based. Anda harus memigrasikan instans ini ke jenis Nitro-based instans sebelum Anda dapat memindahkannya ke Nitro-based Host Khusus.

> **Important**
>
> Sebelum memulai migrasi instans, sebaiknya buat cadangan data. Untuk informasi selengkapnya, lihat Membuat snapshot Amazon EBS multi-volume dari instans Amazon EC2.

Untuk menemukan instans yang berjalan di Host Xen-based Khusus Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih Xen-based host yang ingin Anda migrasikan, lalu pilih tab Instans yang sedang berjalan. Tab mencantumkan semua instance yang berjalan dari host yang dipilih.

Untuk memigr asikan instans Linux, lihatPerubahan jenis instans Amazon EC2.

Untuk memigr asikan instans Windows, lihatMemigrasikan instance EC2 Windows ke jenis Nitro-based instans.

> **Note**
>
> Pastikan Anda memigrasikan instans ke jenis instans yang cocok Nitro-based dengan Host Khusus yang ingin Anda migrasi. Misalnya, jika Anda ingin bermigrasi ke Host Khusus M7i, pastikan Anda memigrasikan instans ke jenis instans M7i.

Langkah 3: Alokasikan Host Nitro-based Khusus

Untuk menemukan Host Nitro-based Khusus yang didukung

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Jenis Instance.
3. Terapkan filter berikut:
   - Hypervisor = nitro
   - Dukungan Host Khusus = benar

Setelah Anda menemukan jenis Nitro-based instans yang sesuai, alokasikan Host Khusus baru.

Langkah 4: Pindahkan instance yang dimigrasi ke Host Nitro-based Khusus baru

Setelah mengalokasikan Host Nitro-based Khusus dan mencapai `available` status, Anda dapat memindahkan instans yang sebelumnya Anda migrasikan ke jenis Nitro-based instans ke Host Khusus yang baru.

Untuk memindahkan instans ke Host Nitro-based Khusus baru

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, Instans.
3. Pilih instance yang Anda migrasikan dan pilih Tindakan, Pengaturan instans, Ubah penempatan instans.
4. Untuk host khusus Target, pilih Host Nitro-based Khusus baru, lalu pilih S impan.
5. Mulai ulang instans. Pilih instance dan pilih Status Instance, Mulai instance.

Langkah 5: Lepaskan Host Xen-based Khusus yang tidak digunakan

Setelah memigrasikan beban kerja dari Host Khus Xen-based us ke Host Nitro-based Khusus yang baru, Anda dapat melepaskan Host Xen-based Khusus jika Anda tidak lagi membutuhkannya.

## Beli Reservasi Host Khusus untuk diskon penagihan Host Khusus

Reservasi Tuan Rumah Khusus memberi Anda diskon hingga 70 persen dibandingkan dengan harga Host On-Demand Khusus. Anda harus memiliki Host Khusus aktif yang dialokasikan di akun Anda sebelum Anda dapat membeli Reservasi Host Khusus. Untuk informasi selengkapnya, lihat Reservasi Host Khusus.

Console

Untuk membeli reservasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Host Khusus, Reservasi Host Khusus, Beli Reservasi Host Khusus.
3. Pada layar Temukan penawaran, lakukan hal berikut:
   a. Untuk keluarga Inst ance, pilih keluarga instans Host Khusus yang akan dibeli Reservasi Host Khusus.
   b. Untuk Opsi Pembay aran, pilih dan konfigurasikan opsi pembayaran pilihan Anda.
4. Pilih Berikutnya.
5. Pilih Host Khusus yang akan dihubungkan dengan Reservasi Host Khusus, lalu pilih Ber ikutnya.
6. (Opsional) Tetapkan tag ke Reservasi Host Khusus.
7. Tinjau pesanan Anda dan pilih Pembelian.

AWS CLI

Untuk membeli reservasi

1. Gunakan perintah describe-host-reservation-offers untuk membuat daftar penawaran yang tersedia yang sesuai dengan kebutuhan Anda. Contoh berikut menampilkan daftar penawaran yang mendukung instans di keluarga instans `m4` dan memiliki jangka waktu satu tahun.

   Jangka waktu ditentukan dalam hitungan detik. Jangka waktu satu tahun mencakup 31.536.000 detik, dan jangka waktu tiga tahun mencakup 94.608.000 detik.

   ```
   aws ec2 describe-host-reservation-offerings \
       --filter Name=instance-family,Values=m4 \
       --max-duration 31536000
   ```

   Perintah menampilkan daftar penawaran yang sesuai dengan kriteria Anda. Catat ID penawaran untuk membeli.

2. Gunakan perintah purchase-host-reservation untuk membeli penawaran dan memberikan yang `offeringId` dicatat pada langkah sebelumnya. Contoh berikut membeli reservasi yang ditentukan dan mengaitkannya dengan Host Khusus tertentu yang sudah dialokasikan di AWS akun, dan menerapkan tag dengan kunci `purpose` dan nilai`production`.

   ```
   aws ec2 purchase-host-reservation \
       --offering-id hro-03f707bf363b6b324 \
       --host-id-set h-013abcd2a00cbd123 \
       --tag-specifications 'ResourceType=host-
   reservation,Tags={Key=purpose,Value=production}'
   ```

PowerShell

Untuk membeli reservasi

1. Gunakan Get-EC2HostReservationOffering cmdlet untuk membuat daftar penawaran yang tersedia yang sesuai dengan kebutuhan Anda. Contoh berikut mencantumkan penawaran yang mendukung instans di keluarga instans `m5` dan memiliki jangka waktu satu tahun.

   Jangka waktu ditentukan dalam hitungan detik. Jangka waktu satu tahun mencakup 31.536.000 detik, dan jangka waktu tiga tahun mencakup 94.608.000 detik.

```
$filter = @{Name="instance-family"; Values="m5"}
Get-EC2HostReservationOffering `
    -Filter $filter `
    -MaxDuration 31536000
```

Perintah menampilkan daftar penawaran yang sesuai dengan kriteria Anda. Catat ID penawaran untuk membeli.

2. Gunakan New-EC2HostReservation cmdlet untuk membeli penawaran dan memberikan ID penawaran yang dicatat pada langkah sebelumnya. Contoh berikut membeli reservasi yang ditentukan dan mengaitkannya dengan Host Khusus tertentu yang sudah dialokasikan dalam AWS akun.

```
New-EC2HostReservation `
    -OfferingId hro-03f707bf363b6b324 `
    -HostIdSet h-013abcd2a00cbd123
```

## Cross-account Berbagi Host Khusus Amazon EC2

Berbagi Host Khusus memungkinkan pemilik Host Khusus untuk membagikan Host Khusus mereka dengan AWS akun lain atau dalam AWS organisasi. Ini memungkinkan Anda untuk membuat dan mengelola Host Khusus secara terpusat, dan berbagi Host Khusus di beberapa AWS akun atau di dalam AWS organisasi Anda.

Dalam model ini, AWS akun yang memiliki Host Khusus (pemilik) membagikannya dengan AWS akun lain (konsumen). Konsumen dapat meluncurkan instans ke Host Khusus yang dibagikan dengan mereka dengan cara yang sama seperti saat meluncurkan instans ke Host Khusus yang mereka alokasikan di akun mereka sendiri. Pemilik bertanggung jawab untuk mengelola Host Khusus dan instans yang mereka luncurkan ke dalamnya. Pemilik tidak dapat memodifikasi instans yang diluncurkan konsumen ke Host Khusus bersama. Konsumen bertanggung jawab untuk mengelola instans yang mereka luncurkan ke Host Khusus yang dibagikan dengan mereka. Konsumen tidak dapat melihat atau memodifikasi instans yang dimiliki oleh konsumen lain atau oleh pemilik Host Khusus, dan mereka tidak dapat memodifikasi Host Khusus yang dibagikan dengan mereka.

Pemilik Host Khusus dapat berbagi Host Khusus dengan:

- AWS Akun tertentu di dalam atau di luar AWS organisasinya
- Unit organisasi di dalam AWS organisasinya

- Seluruh AWS organisasinya

Daftar Isi


- Prasyarat untuk berbagi Host Khusus
- Batasan untuk berbagi Host Khusus
- Layanan terkait
- Berbagi di seluruh Zona Ketersediaan
- Izin Host Khusus Bersama
- Tagihan dan pengukuran
- Batas Host Khusus
- Pemulihan host dan berbagi Host Khusus
- Bagikan Host Khusus Amazon EC2 di seluruh AWS rekening
- Batalkan berbagi Host Khusus yang dibagikan dengan orang lain AWS rekening
- Lihat Host Khusus Amazon EC2 yang dibagikan di AWS akun


## Prasyarat untuk berbagi Host Khusus

- Untuk berbagi Host Khusus, Anda harus memilikinya di AWS akun Anda. Anda tidak dapat berbagi Host Khusus yang telah dibagikan dengan Anda.
- Untuk berbagi Host Khusus dengan AWS organisasi atau unit organisasi di AWS organisasi Anda, Anda harus mengaktifkan berbagi dengan AWS Organizations. Untuk informasi selengkapnya, lihat Aktifkan Berbagi dengan AWS Organizations dalam Panduan Pengguna AWS RAM .

## Batasan untuk berbagi Host Khusus

Anda tidak dapat membagikan Host Khusus yang telah dialokasikan untuk tipe instans berikut: `u-6tb1.metal`, `u-9tb1.metal`, `u-12tb1.metal`, `u-18tb1.metal`, dan `u-24tb1.metal`.

## Layanan terkait

### AWS Resource Access Manager

Berbagi Host Khusus terintegrasi dengan AWS Resource Access Manager (AWS RAM). AWS RAM adalah layanan yang memungkinkan Anda untuk berbagi AWS sumber daya Anda dengan AWS

akun apa pun atau melalui AWS Organizations. Dengan AWS RAM, Anda berbagi sumber daya yang Anda miliki dengan membuat berbagi sumber daya. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan dibagikan. Konsumen dapat berupa AWS akun individu, atau unit organisasi atau seluruh organisasi dari AWS Organizations.

Untuk informasi selengkap AWS RAM nya, lihat Panduan AWS RAM Pengguna .

Berbagi di seluruh Zona Ketersediaan

Untuk memastikan bahwa sumber daya didistribusikan di seluruh Zona Ketersediaan untuk Wilayah, Zona Ketersediaan dipetakan secara independen ke nama untuk setiap akun. Hal ini dapat menyebabkan perbedaan penamaan Zona Ketersediaan di seluruh akun. Misalnya, Zona Ketersediaan `us-east-1a` untuk AWS akun Anda mungkin tidak memiliki lokasi yang sama `us-east-1a` dengan AWS akun lain.

Untuk mengidentifikasi lokasi Host Khusus Anda yang terkait dengan akun Anda, Anda harus menggunakan ID Zona Ketersediaan (AZ ID). ID Zona Ketersediaan adalah pengenal unik dan konsisten untuk Zona Ketersediaan di semua AWS akun. Misalnya, `use1-az1` adalah ID Zona Ketersediaan untuk Wilayah `us-east-1` dan lokasinya sama di setiap akun AWS .

Untuk melihat ID Zona Ketersediaan untuk Zona Ketersediaan di akun Anda

1. Buka AWS RAM konsol di https://console.aws.amazon.com/ram/home.
2. ID Zona Ketersediaan untuk Wilayah saat ini ditampilkan di panel ID AZ Anda di sisi kanan layar.

Izin Host Khusus Bersama

Izin untuk pemilik

Pemilik bertanggung jawab untuk mengelola Host Khusus bersama dan instans yang mereka luncurkan ke dalamnya. Pemilik dapat melihat semua instans yang berjalan di Host Khusus bersama, termasuk yang diluncurkan oleh konsumen. Namun, pemilik tidak dapat mengambil tindakan apa pun untuk menjalankan instans yang diluncurkan oleh konsumen.

Izin untuk konsumen

Konsumen bertanggung jawab untuk mengelola instans yang mereka luncurkan ke Host Khusus bersama. Konsumen tidak dapat mengubah Host Khusus bersama dengan cara apa pun, dan mereka tidak dapat melihat atau memodifikasi instans yang diluncurkan oleh konsumen lain atau pemilik Host Khusus.

Tagihan dan pengukuran

Tidak ada biaya tambahan untuk berbagi Host Khusus.

Pemilik ditagih untuk Host Khusus yang mereka bagikan. Konsumen tidak akan ditagih untuk instans yang mereka luncurkan ke Host Khusus bersama.

Reservasi Host Khusus terus memberikan diskon penagihan untuk Host Khusus bersama. Hanya pemilik Host Khusus yang dapat membeli Reservasi Host Khusus untuk Host Khusus bersama yang mereka miliki.

Batas Host Khusus

Host Khusus Bersama dihitung dalam batas Host Khusus pemilik saja. Batas Host Khusus konsumen tidak terpengaruh oleh Host Khusus yang telah dibagikan dengan mereka. Demikian pula, instans yang diluncurkan konsumen ke Host Khusus bersama tidak diperhitungkan dalam batas instans mereka.

Pemulihan host dan berbagi Host Khusus

Pemulihan host memulihkan instans yang diluncurkan oleh pemilik Host Khusus dan konsumen yang telah membagikannya. Host Khusus pengganti dialokasikan ke akun pemilik. Ini ditambahkan ke sumber daya yang sama dengan Host Khusus asli, dan dibagikan dengan konsumen yang sama.

Untuk informasi selengkapnya, lihat Pemulihan Host Khusus Amazon EC2.

Bagikan Host Khusus Amazon EC2 di seluruh AWS rekening

Saat pemilik membagikan Host Khusus, konsumen akan dapat meluncurkan instans di host. Konsumen dapat meluncurkan sebanyak mungkin instans ke host bersama sesuai kapasitas yang tersedia.

> **Important**
>
> Perhatikan bahwa Anda bertanggung jawab untuk memastikan bahwa Anda memiliki hak lisensi yang sesuai untuk membagikan lisensi BYOL apa pun pada Host Khusus Anda.

Jika Anda berbagi Host Khusus dengan penempatan otomatis diaktifkan, perhatikan hal berikut karena dapat menyebabkan penggunaan Host Khusus yang tidak diinginkan:

- Jika konsumen meluncurkan instans dengan penghunian Host Khusus dan mereka tidak memiliki kapasitas pada Host Khusus yang mereka miliki di akun mereka, instans tersebut secara otomatis diluncurkan ke Host Khusus bersama.

Untuk membagikan Host Khusus, Anda harus menambahkannya ke berbagi sumber daya. Berbagi sumber daya adalah AWS RAM sumber daya yang memungkinkan Anda berbagi sumber daya di seluruh AWS akun. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan berbagi dengan mereka. Anda dapat menambahkan Host Khusus ke sumber daya yang ada, atau Anda dapat menambahkannya ke berbagi sumber daya baru.

Jika Anda adalah bagian dari organisasi AWS Organizations dan berbagi dalam organisasi Anda diaktifkan, konsumen di organisasi Anda secara otomatis diberikan akses ke Host Khusus bersama. Jika tidak, konsumen menerima undangan untuk bergabung dengan berbagi sumber daya dan diberikan akses ke Host Khusus bersama setelah menerima undangan.

> **Note**
>
> Setelah Anda membagikan Host Khusus, konsumen mungkin perlu waktu beberapa menit untuk dapat mengaksesnya.

Console

Untuk membagikan Host Khusus yang Anda miliki menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih Host Khusus untuk berbagi dan pilih Tindakan, Bagikan host.
4. Pilih berbagi sumber daya yang ingin ditambahkan Host Khusus dan pilih Bagikan host.

   Butuh beberapa menit bagi konsumen untuk mendapatkan akses ke host bersama.

Untuk berbagi Host Khusus yang Anda miliki menggunakan AWS RAM konsol

Lihat Membuat berbagi sumber daya di Panduan AWS RAM Pengguna.

AWS CLI

Untuk berbagi Host Khusus yang Anda miliki

Gunakan perintah create-resource-share.

```
aws ram create-resource-share \
    --name my-resource-share \
    --resource-arns arn:aws:ec2:us-east-2:123456789012:dedicated-
host/h-07879acf49EXAMPLE
```

PowerShell

Untuk berbagi Host Khusus yang Anda miliki

Gunakan New-RAMResourceShare cmdlet.

```
New-RAMResourceShare `
    -Name my-resource-share `
    -ResourceArn arn:aws:ec2:us-east-2:123456789012:dedicated-
host/h-07879acf49EXAMPLE
```

Batalkan berbagi Host Khusus yang dibagikan dengan orang lain AWS rekening

Pemilik Host Khusus dapat membatalkan pembagian Host Khusus bersama kapan saja. Saat Anda membatalkan berbagi Host Khusus bersama, aturan berikut ini berlaku:

- Konsumen yang dibagikan Host Khusus tidak lagi dapat meluncurkan instans baru ke dalamnya.
- Instans yang dimiliki oleh konsumen yang berjalan pada Host Khusus pada waktu pembatalan pembagian terus berjalan, tetapi dijadwalkan untuk pensiun. Konsumen menerima notifikasi pensiun untuk instans tersebut dan mereka memiliki waktu dua minggu untuk mengambil tindakan atas notifikasi tersebut. Namun, jika Host Khusus dibagikan ulang dengan konsumen dalam periode pemberitahuan pensiun, pensiun instans dibatalkan.

Untuk membatalkan berbagi Host Khusus bersama yang Anda miliki, Anda harus menghapusnya dari berbagi sumber daya.

Console

Untuk membatalkan berbagi Host Khusus bersama yang Anda miliki

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.

3. Pilih Host Khusus yang batal dibagikan dan pilih tab Berbagi.
4. Tab Berbagi mencantumkan sumber daya yang telah ditambahkan Host Khusus. Pilih bagian sumber daya untuk menghapus Host Khusus dan pilih Hapus host dari berbagi sumber daya.

Untuk membatalkan berbagi Host Khusus bersama yang Anda miliki menggunakan AWS RAM konsol

Lihat Memperbarui berbagi sumber daya di Panduan AWS RAM Pengguna.

AWS CLI

Untuk membatalkan berbagi Host Khusus bersama yang Anda miliki

Gunakan perintah disassociate-resource-share.

```
aws ram disassociate-resource-share \
    --resource-share-arn arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE \
 --resource-arns arn:aws:ec2:us-east-2:123456789012:dedicated-host/h-07879acf49EXAMPLE
```

PowerShell

Untuk membatalkan berbagi Host Khusus bersama yang Anda miliki

Gunakan Disconnect-RAMResourceShare cmdlet.

```
Disconnect-RAMResourceShare `
    -ResourceShareArn "arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE" `
    -ResourceArn "arn:aws:ec2:us-east-2:123456789012:dedicated-host/h-07879acf49EXAMPLE"
```

Lihat Host Khusus Amazon EC2 yang dibagikan di AWS akun

Anda dapat melihat Host Khusus yang Anda bagikan dengan akun lain, dan Host Khusus yang dibagikan dengan Anda. Jika Anda memiliki Host Khusus, Anda dapat melihat semua instans yang berjalan di host, termasuk instans yang diluncurkan oleh konsumen. Jika Host Khusus dibagikan dengan Anda, Anda hanya dapat melihat instance yang Anda luncurkan ke host bersama, dan bukan yang diluncurkan oleh konsumen lain.

Pemilik dan konsumen dapat mengidentifikasi Host Khusus bersama menggunakan salah satu metode berikut.

Console

Untuk mengidentifikasi Host Khusus bersama

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus. Layar mencantumkan Host Khusus yang Anda miliki dan Host Khusus yang dibagikan dengan Anda.
3. Kolom Pemilik menunjukkan ID akun AWS dari pemilik Host Khusus.
4. Untuk melihat instance yang berjalan di host, pilih tab Instans.

AWS CLI

Untuk mengidentifikasi Host Khusus bersama

Gunakan perintah describe-host. Perintah tersebut menampilkan Host Khusus yang Anda miliki dan Host Khusus yang dibagikan dengan Anda. Nilai `Owner` adalah ID akun pemilik Host Khusus. `Instances`Daftar ini menjelaskan instance yang berjalan di host.

```
aws ec2 describe-hosts --filter "Name=state,Values=available"
```

PowerShell

Untuk mengidentifikasi Host Khusus bersama

Gunakan Get-EC2host cmdlet. Cmdlet mengembalikan Host Khusus yang Anda miliki dan Host Khusus yang dibagikan dengan Anda. Nilai `Owner` dalam respons adalah ID akun pemilik Host Khusus. `Instances`Daftar ini menjelaskan instance yang berjalan di host.

```
Get-EC2Host -Filter @{Name="state"; Values="available"}
```

## Host Khusus Amazon EC2 aktif AWS Outposts

AWS Outposts adalah layanan yang dikelola sepenuhnya yang memperluas AWS infrastruktur, layanan, API, dan alat ke tempat Anda. Dengan menyediakan akses lokal ke infrastruktur AWS terkelola, AWS Outposts memungkinkan Anda membangun dan menjalankan aplikasi di tempat

menggunakan antarmuka pemrograman yang sama seperti di Wil AWS ayah, sambil menggunakan sumber daya komputasi dan penyimpanan lokal untuk latensi yang lebih rendah dan kebutuhan pemrosesan data lokal.

Outpost adalah kumpulan kapasitas AWS komputasi dan penyimpanan yang digunakan di situs pelanggan. AWS mengoperasikan, memantau, dan mengelola kapasitas ini sebagai bagian dari Wil AWS ayah.

Anda dapat mengalokasikan Host Khusus di Outposts yang Anda miliki di akun Anda. Ini memudahkan Anda untuk membawa lisensi perangkat lunak dan beban kerja yang ada yang memerlukan server fisik khusus ke AWS Outposts. Anda juga dapat menargetkan aset perangkat keras tertentu di Outpost untuk membantu meminimalkan latensi di antara beban kerja Anda.

Host Khusus memungkinkan Anda untuk menggunakan lisensi perangkat lunak yang memenuhi syarat di Amazon EC2, sehingga Anda mendapatkan fleksibilitas dan efektivitas biaya menggunakan lisensi Anda sendiri. Lisensi perangkat lunak lain yang terikat pada mesin virtual, soket, atau inti fisik, juga dapat digunakan pada Host Khusus, tunduk pada persyaratan lisensi mereka. Meskipun Outposts selalu menjadi lingkungan penghunian tunggal yang memenuhi syarat untuk beban kerja BYOL, Host Khusus memungkinkan Anda membatasi lisensi yang diperlukan untuk satu host dibandingkan dengan deployment seluruh Outpost.

Selain itu, menggunakan Host Khusus di Outpost memberi Anda fleksibilitas yang lebih besar dalam deployment tipe instans, dan kontrol yang lebih terperinci atas penempatan instans. Anda dapat menargetkan host tertentu untuk peluncuran instans dan menggunakan afinitas host untuk memastikan bahwa instans selalu berjalan pada host tersebut, atau Anda dapat menggunakan penempatan otomatis untuk meluncurkan instans ke host mana pun yang tersedia yang memiliki konfigurasi dan ketersediaan kapasitas yang cocok.

Daftar Isi


- Prasyarat
- Fitur yang didukung
- Pertimbangan
- Alokasikan Host Khusus Amazon EC2 pada AWS Outposts


## Prasyarat

Anda harus menginstal Outpost di situs Anda. Untuk informasi selengkapnya, lihat Membuat Outpost dan memesan kapasitas Outpost di Panduan Pengguna AWS Outposts .

### Fitur yang didukung

- Keluarga instance berikut didukung:
  - Tujuan umum: M5 | M5d | M7i | M8i
  - Komputasi dioptimalkan: C5 | C5d | C7i | C8i
  - Memori dioptimalkan: R5 | R5d | R7i | R8i
  - Penyimpanan dioptimalkan: i3en
  - Komputasi yang dipercepat: G4dn
- Host Khusus di Outposts dapat dikonfigurasi untuk mendukung beberapa ukuran instans. Dukungan untuk beberapa ukuran instans tersedia untuk keluarga instans berikut.
  - Tujuan umum: M5 | M5d | M7i
  - Komputasi dioptimalkan: C5 | C5d | C7i
  - Memori dioptimalkan: R5 | R5d | R7i

  Untuk informasi selengkapnya, lihat Konfigurasi kapasitas instans Host Khusus Amazon EC2.
- Host Khusus di Outposts mendukung penempatan otomatis dan peluncuran instans tertarget. Untuk informasi selengkapnya, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.
- Host Khusus di Outposts mendukung afinitas host. Untuk informasi selengkapnya, lihat Penempatan otomatis Host Khusus Amazon EC2 dan afinitas host.
- Host Khusus di Outposts mendukung berbagi dengan AWS RAM. Untuk informasi selengkapnya, lihat Cross-account Berbagi Host Khusus Amazon EC2.

### Pertimbangan

- Reservasi Host Khusus tidak didukung di Outposts.
- Tuan rumah grup sumber daya dan tidak AWS License Manager didukung di Outpost.
- Host Khusus di Outposts tidak mendukung instans T3 yang dapat melonjak.
- Host Khusus di Outposts tidak mendukung pemulihan host.
- Pemulihan otomatis yang disederhanakan tidak didukung untuk instans dengan penyewaan Host Khusus di Pos Terdepan.

## Alokasikan Host Khusus Amazon EC2 pada AWS Outposts

Anda mengalokasikan dan menggunakan Host Khusus di Outposts dengan cara yang sama dengan Host Khusus di Wilayah AWS .

### Prasyarat

Buat subnet pada Outpost. Untuk informasi selengkapnya, lihat Membuat subnet di Panduan Pengguna AWS Outposts .

Untuk mengalokasikan Host Khusus di Outpost, gunakan salah satu metode berikut:

### Console

Untuk mengalokasikan Host Khusus di Pos Terdepan menggunakan AWS Outposts konsol

1. Buka AWS Outposts konsol di https://console.aws.amazon.com/outposts/.
2. Di panel navigasi, pilih Outposts. Pilih Outpost kemudian pilih Tindakan, Alokasikan Host Khusus.
3. Konfigurasikan Host Khusus sesuai kebutuhan. Untuk informasi selengkapnya, lihat Mengalokasikan Host Khusus Amazon EC2 untuk digunakan di akun Anda.

   > **Note**
   >
   > Zona Ketersediaan dan ARN Outpost harus diisi sebelumnya dengan Zone Ketersediaan dan ARN dari Outpost yang dipilih.

4. Pilih Alokasikan.

Untuk mengalokasikan Host Khusus di Outpost menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus, lalu pilih Alokasi Host Khusus.
3. Untuk Zona Ketersediaan, pilih Zona Ketersediaan yang terkait dengan Outpost.
4. Untuk ARN Outpost masukkan ARN Outpost.
5. Untuk menargetkan aset perangkat keras tertentu di Outpost, pada Menargetkan aset perangkat keras tertentu di Outpost, pilih Aktifkan. Untuk setiap aset perangkat keras yang ditargetkan, pilih Tambahkan ID aset, lalu masukkan ID aset perangkat keras.

> **Note**
>
> Nilai yang Anda tentukan untuk Kuantitas harus sama dengan jumlah ID aset yang Anda tentukan. Misalnya, jika Anda menentukan 3 ID aset, maka Kuantitas juga harus 3.

6. Konfigurasikan pengaturan Host Khusus yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat Mengalokasikan Host Khusus Amazon EC2 untuk digunakan di akun Anda.
7. Pilih Alokasikan.

AWS CLI

Untuk mengalokasikan Host Khusus di Pos Terdepan

Gunakan perintah allocate-hosts. Untuk `--availability-zone`, tentukan Zona Ketersediaan yang terkait dengan Outpost. Untuk `--outpost-arn`, tentukan ARN dari Outpost. Secara opsional, untuk `--asset-ids`, tentukan ID aset perangkat keras Outpost yang ditargetkan.

```
aws ec2 allocate-hosts \
    --availability-zone "us-east-1a" \
    --outpost-arn "arn:aws:outposts:us-east-1a:111122223333:outpost/
op-4fe3dc21baEXAMPLE" \
    --asset-ids asset_id \
    --instance-family "m5" \
    --auto-placement "off" \
    --quantity 1
```

PowerShell

Untuk mengalokasikan Host Khusus di Pos Terdepan

Gunakan New-EC2Host cmdlet. Tentukan Zona Ketersediaan yang terkait dengan Pos Terdepan. Secara opsional, untuk `-AssetId`, tentukan ID aset perangkat keras Outpost yang ditargetkan.

```
New-EC2Host `
    -AvailabilityZone "us-east-1a" `
    -OutpostArn "arn:aws:outposts:us-east-1a:111122223333:outpost/
op-4fe3dc21baEXAMPLE" `
```

```
    -AssetId asset_id `
    -InstanceFamily "m5" `
    -AutoPlacement "off" `
    -Quantity 1
```

Untuk meluncurkan sebuah instans ke Host Khusus di Outpost

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus. Pilih Host Khusus yang Anda alokasikan pada langkah sebelumnya dan pilih Actions, Launch instans ke host.
3. Konfigurasikan instans sesuai kebutuhan kemudian luncurkan instans. Untuk informasi selengkapnya, lihat Luncurkan instans Amazon EC2 di Host Khusus Amazon EC2.

## Pemulihan Host Khusus Amazon EC2

Pemulihan otomatis Host Khusus memulai ulang instans Anda ke host pengganti baru saat kondisi bermasalah tertentu terdeteksi di Host Khusus Anda. Pemulihan host mengurangi kebutuhan akan intervensi manual dan menurunkan beban operasional jika ada kegagalan Host Khusus yang tidak terduga terkait daya sistem atau peristiwa konektivitas jaringan. Masalah Host Khusus lainnya akan memerlukan intervensi manual dalam pemulihannya.

Daftar Isi


- Cara kerja pemulihan Host Khusus Amazon EC2
- Jenis instans Supportyang didukung
- Harga
- Kelola pemulihan Host Khusus Amazon EC2
- Melihat pengaturan pemulihan host untuk Host Khusus Amazon EC2 Anda
- Memulihkan instans secara manual yang tidak didukung oleh pemulihan Host Khusus Amazon EC2


### Cara kerja pemulihan Host Khusus Amazon EC2

Host Khusus dan proses pemulihan grup sumber daya host menggunakan pemeriksaan kondisi tingkat host untuk menilai ketersediaan Host Khusus dan untuk mendeteksi kegagalan sistem dasar. Tipe kegagalan Host Khusus menentukan apakah pemulihan otomatis Host Khusus dimungkinkan. Contoh masalah yang dapat menyebabkan pemeriksaan kondisi tingkat host gagal meliputi:

- Hilangnya konektivitas jaringan
- Hilangnya daya sistem
- Masalah perangkat keras atau perangkat lunak pada host fisik

> **Important**
>
> Pemulihan otomatis Host Khusus tidak terjadi ketika host dijadwalkan pensiun.

Pemulihan otomatis Host Khusus

Ketika kegagalan daya sistem atau konektivitas jaringan terdeteksi pada Host Khusus Anda, pemulihan otomatis Host Khusus dimulai dan Amazon EC2 secara otomatis mengalokasikan Host Khusus pengganti di Zona Ketersediaan yang sama dengan Host Khusus asli. Host Khusus pengganti menerima ID host baru, tetapi mempertahankan atribut yang sama dengan Host Khusus yang asli, termasuk:

- Zona Ketersediaan
- Jenis instans
- Tag
- Pengaturan penempatan otomatis
- Reservasi

Saat Host Khusus pengganti dialokasikan, instans dipulihkan ke Host Khusus pengganti. Instans yang dipulihkan mempertahankan atribut yang sama dengan instans asli, termasuk:

- ID Instans
- Alamat IP privat
- Alamat IP elastic
- Lampiran volume EBS
- Semua metadata instans

Selain itu, integrasi bawaan dengan Manajer AWS Lisensi mengotomatiskan pelacakan dan pengelolaan lisensi Anda.

> **Note**
>
> AWS Integrasi Manajer Lisensi hanya didukung di Wilayah di mana Manajer AWS Lisensi tersedia.

Jika instans memiliki hubungan afinitas host dengan Host Khusus yang terganggu, instans yang dipulihkan membentuk afinitas host dengan Host Khusus pengganti.

Jika semua instans telah dipulihkan ke Host Khusus pengganti, Host Khusus yang terganggu akan dilepas, dan Host Khusus pengganti tersedia untuk digunakan.

Saat pemulihan host dimulai, pemilik AWS akun diberitahu melalui email dan Dasbor AWS Health acara. Notifikasi kedua dikirimkan setelah pemulihan host berhasil diselesaikan.

Jika Anda menggunakan Manajer AWS Lisensi untuk melacak lisensi Anda, Manajer AWS Lisensi mengalokasikan lisensi baru untuk Host Khusus pengganti berdasarkan batas konfigurasi lisensi. Jika konfigurasi lisensi memiliki batasan keras yang akan dilanggar sebagai akibat dari pemulihan host, proses pemulihan tidak diizinkan dan Anda diberitahu tentang kegagalan pemulihan host melalui pemberitahuan Amazon SNS (jika pengaturan notifikasi telah dikonfigurasi untuk Manajer AWS Lisensi). Jika konfigurasi lisensi memiliki batas lunak yang akan dilanggar sebagai akibat dari pemulihan host, pemulihan diizinkan untuk dilanjutkan dan Anda diberi tahu tentang pelanggaran batas melalui notifikasi Amazon SNS. Untuk informasi selengkapnya, lihat Menggunakan Konfigurasi Lisensi dan Pengaturan di License Manager di Panduan Pengguna AWS License Manager.

Status pemulihan host

Saat kegagalan Host Khusus terdeteksi, Host Khusus yang terganggu memasuki status `under-assessment`, dan semua instance masuk ke status `impaired`. Anda tidak dapat meluncurkan instans ke Host Khusus yang rusak saat berada dalam status `under-assessment`.

Setelah Host Khusus pengganti dialokasikan, host memasuki status `pending`. Statusnya tidak berubah sampai proses pemulihan host selesai. Anda tidak dapat meluncurkan instans ke Host Khusus pengganti saat berada dalam status `pending`. Instans yang dipulihkan pada Host Khusus pengganti tetap berada dalam status `impaired` selama proses pemulihan.

Setelah pemulihan host selesai, Host Khusus pengganti memasuki status `available`, dan instans yang dipulihkan kembali ke status `running`. Anda dapat meluncurkan instans ke Host Khusus pengganti setelah instans berada dalam status `available`. Host Khusus yang mengalami gangguan dilepas secara permanen dan masuk dalam status `released-permanent-failure`.

Jika Host Khusus yang rusak memiliki instans yang tidak mendukung pemulihan host, seperti instans dengan volume root penyimpanan instans, Host Khusus tidak dirilis. Sebaliknya, host ditandai untuk pensiun dan memasuki status `permanent-failure`.

Skenario tanpa pemulihan otomatis Host Khusus

Pemulihan otomatis Host Khusus tidak terjadi ketika host dijadwalkan pensiun. Anda akan menerima pemberitahuan pensiun di Dasbor AWS Health CloudWatch acara Amazon, dan alamat email pemilik AWS akun menerima pesan mengenai kegagalan Host Khusus. Ikuti langkah-langkah perbaikan yang dijelaskan dalam notifikasi pensiun dalam jangka waktu yang ditentukan untuk secara manual memulihkan instans pada host yang pensiun.

Instans yang dihentikan tidak dipulihkan ke Host Khusus pengganti. Jika Anda mencoba untuk memulai contoh instans yang berhenti yang menargetkan Host Khusus yang terganggu, instans akan mulai gagal. Kami menyarankan Anda mengubah instans yang dihentikan untuk menargetkan Host Khusus yang berbeda, atau untuk meluncurkan pada Host Khusus apa pun yang tersedia dengan konfigurasi yang cocok dan penempatan otomatis diaktifkan.

Instans dengan penyimpanan instans tidak dipulihkan ke Host Khusus pengganti. Sebagai langkah perbaikan, Host Khusus yang mengalami gangguan ditandai untuk pensiun dan Anda akan menerima notifikasi pensiun setelah pemulihan host selesai. Ikuti langkah-langkah perbaikan yang dijelaskan dalam notifikasi pensiun dalam jangka waktu yang ditentukan untuk secara manual memulihkan instans yang tersisa pada Host Khusus yang terganggu.

Jenis instans Supportyang didukung

Pemulihan host didukung untuk keluarga contoh berikut:

- Tujuan umum:  A1 | M3 | M4 | M5 | M5n | M5Zn | M6a | M6g | M6i | T3 | Mac1 | Mac2 | | | | M6in | M7a | M7g | M7i | | | Mac2-m1ultra Mac2-m2 Mac2-m2pro Mac-m4 Mac-m4pro Mac-m4max Mac-m3ultra
- Komputasi dioptimalkan:  C3 | C4 | C5 | C5n | C6a | C6g | C6i | C6gN | C6in | C7g | C7gN | C7i
- Memori dioptimalkan:  R3 | R4 | R5 | R5b | R5n | R6g | R6i | | | | | X1 | X1e | X2ieZn | R6a U-6tb1 | R6 U-9tb1 in U-12tb1 | R7g U-18tb1 U-24tb1 | R7i | R7iz
- Komputasi yang dipercepat:  Inf1 | G3 | G5g | P3

Untuk memulihkan instans yang tidak didukung, lihat Memulihkan instans secara manual yang tidak didukung oleh pemulihan Host Khusus Amazon EC2.

> **Note**
>
> Pemulihan otomatis Host khusus untuk tipe instans metal yang didukung akan membutuhkan waktu lebih lama untuk mendeteksi dan memulihkan dari tipe instans nonmetal.

Harga

Tidak ada biaya tambahan untuk menggunakan pemulihan host, tetapi biaya Host Khusus yang biasa berlaku. Untuk informasi selengkapnya, lihat Harga Host Khusus Amazon EC2.

Segera setelah pemulihan host dimulai, Anda tidak lagi ditagih untuk Host Khusus yang terganggu. Tagihan untuk Host Khusus pengganti dimulai hanya setelah host masuk dalam status `available`.

Jika Host Khusus yang mengalami gangguan ditagih menggunakan On-Demand tarif, Host Khusus pengganti juga ditagih menggunakan tarif tersebut On-Demand . Jika Host Khusus yang mengalami gangguan memiliki Reservasi Host Khusus yang aktif, Host Khusus tersebut akan ditransfer ke Host Khusus pengganti.

Kelola pemulihan Host Khusus Amazon EC2

Pemulihan otomatis Host Khusus memulai ulang instans Anda ke host pengganti baru saat kondisi bermasalah tertentu terdeteksi di Host Khusus Anda. Anda dapat mengaktifkan pemulihan host saat mengalokasikan Host Khusus atau setelah alokasi.

Gunakan prosedur berikut untuk mengaktifkan pemulihan host saat mengalokasikan host.

Console

Untuk mengaktifkan pemulihan host pada alokasi

Saat mengalokasikan Host Khusus menggunakan konsol Amazon EC2, untuk pemulihan Host, pilih Aktifkan. Untuk informasi selengkapnya, lihat Mengalokasikan Host Khusus Amazon EC2 untuk digunakan di akun Anda.

AWS CLI

Untuk mengaktifkan pemulihan host pada alokasi

Gunakan perintah allocate-hosts.

```
aws ec2 allocate-hosts \
```

```
    --instance-type m5.large \
    --availability-zone eu-west-1a \
    --auto-placement on \
    --host-recovery on \
    --quantity 1
```

PowerShell

Untuk mengaktifkan pemulihan host pada alokasi

Gunakan New-EC2Host cmdlet.

```
New-EC2Host `
    -InstanceType m5.large `
    -AvailabilityZone eu-west-1a `
    -AutoPlacement on `
    -HostRecovery on `
    -Quantity 1
```

Gunakan prosedur berikut untuk mengelola pemulihan host untuk Host Khusus.

Console

Untuk mengelola pemulihan host setelah alokasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih host khusus.
4. Pilih T ind akan, Ubah host.
5. Untuk pemulihan Host, pilih atau hapus Aktifkan.
6. Pilih Simpan.

AWS CLI

Untuk mengaktifkan pemulihan host setelah alokasi

Gunakan perintah modify-hosts.

```
aws ec2 modify-hosts \
```

```
    --host-recovery on \
    --host-ids h-012a3456b7890cdef
```

Untuk menonaktifkan pemulihan host setelah alokasi

Gunakan perintah modify-hosts dan tentukan `host-recovery` parameter dengan nilai. `off`

```
aws ec2 modify-hosts \
    --host-recovery off \
    --host-ids h-012a3456b7890cdef
```

PowerShell

Untuk mengaktifkan pemulihan host setelah alokasi

Gunakan Edit-host cmdlet.

```
Edit-EC2Host `
    -HostRecovery on `
    -HostId h-012a3456b7890cdef
```

Untuk menonaktifkan pemulihan host setelah alokasi

Gunakan Edit-EC2Host cmdlet.

```
Edit-EC2Host `
    -HostRecovery off `
    -HostId h-012a3456b7890cdef
```

Melihat pengaturan pemulihan host untuk Host Khusus Amazon EC2 Anda

Anda dapat melihat konfigurasi pemulihan host untuk Host Khusus kapan saja.

Console

Untuk melihat konfigurasi pemulihan host untuk Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih Host Khusus, dan di tab Deskripsi, tinjau bidang Pemulihan Host.

AWS CLI

Untuk melihat konfigurasi pemulihan host untuk Host Khusus

Gunakan perintah describe-host.

```
aws ec2 describe-hosts \
    --host-ids h-012a3456b7890cdef \
    --query Hosts[].HostRecovery
```

Berikut ini adalah output contoh.

```
on
```

PowerShell

Untuk melihat konfigurasi pemulihan host untuk Host Khusus

Gunakan Get-EC2Host cmdlet.

```
(Get-EC2Host -HostId h-012a3456b7890cdef).Hosts | Select HostRecovery
```

Berikut ini adalah output contoh.

```
HostRecovery
------------
on
```

Memulihkan instans secara manual yang tidak didukung oleh pemulihan Host Khusus Amazon EC2

Pemulihan host tidak mendukung pemulihan instans yang menggunakan volume penyimpanan instans. Ikuti petunjuk di bawah ini untuk secara manual memulihkan semua instans Anda yang tidak dapat dipulihkan secara otomatis.

> **Warning**
>
> Data pada volume penyimpanan instans hilang saat instans dihentikan, dihibernasi, atau diakhiri. Ini termasuk volume penyimpanan instans yang dilampirkan ke instance yang memiliki volume root EBS. Untuk melindungi data dari volume penyimpanan instans, cadangkan ke penyimpanan persisten sebelum instans dihentikan atau diakhiri.

Memulihkan EBS-backed instans secara manual

Untuk EBS-backed instans yang tidak dapat dipulihkan secara otomatis, sebaiknya hentikan dan memulai instans secara manual untuk memulihkannya ke Host Khusus baru. Untuk informasi selengkapnya tentang menghentikan instans Anda, dan tentang perubahan yang terjadi dalam konfigurasi instans Anda saat dihentikan, lihat Menghentikan dan memulai instans Amazon EC2.

Memulihkan instans secara manual dengan volume root penyimpanan instance

Untuk contoh dengan volume root penyimpanan instans yang tidak dapat dipulihkan secara otomatis, sebaiknya lakukan hal berikut:

1. Luncurkan instans pengganti pada Host Khusus baru dari AMI terbaru Anda.
2. Migrasikan semua data yang diperlukan ke instans pengganti.
3. Akhiri instans asli pada Host Khusus yang terganggu.

## Pemeliharaan host untuk Host Khusus Amazon EC2

Dengan pemeliharaan host, jika Host Khusus terdegradasi, Amazon EC2 secara otomatis memigrasikan instans yang berjalan di dalamnya ke Host Khusus pengganti yang sehat. Ini membantu meminimalkan waktu henti untuk beban kerja Anda, dan menyederhanakan pengelolaan Host Khusus Anda. Pemeliharaan host juga dilakukan untuk pemeliharaan Amazon EC2 terencana dan rutin.

Amazon EC2 mendukung dua jenis pemeliharaan host:

- Pemeliharaan host migrasi langsung — Instans secara otomatis dimigrasi ke host pengganti dalam waktu 24 jam, tanpa menghentikan dan memulai ulang.
- Reboot-based pemeliharaan host — Instans dijadwalkan misalnya reboot acara ter jadwal, di mana mereka secara otomatis dihentikan dan dimulai ulang pada host pengganti.

Daftar Isi


- Pemeliharaan host versus pemulihan host
- Pertimbangan-pertimbangan
- Layanan terkait
- Harga

- Cara kerja pemeliharaan host untuk Amazon EC2 Dedicated Host
- Mengonfigurasi pengaturan pemeliharaan host untuk Host Khusus Amazon EC2

### Pemeliharaan host versus pemulihan host

Tabel berikut menunjukkan perbedaan utama antara pemulihan host dan pemeliharaan host.

| | Pemulihan host | Pemeliharaan host |
|---|---|---|
| Jangkauan instans | Tidak dapat dijangkau | Dapat dijangkau |
| Status Tuan Rumah Khusus | `under-assessment` | `permanent-failure` |
| Grup sumber daya host | Didukung | Tidak didukung |

Untuk informasi selengkapnya tentang pemulihan host, lihat Pemulihan host.

### Pertimbangan-pertimbangan

- Pemeliharaan host tersedia di semua Wilayah AWS, kecuali Wilayah China dan AWS GovCloud (US) Regions.
- Pemeliharaan host tidak didukung di AWS Outposts, Zona AWS Lokal, dan Zona AWS Panjang Gelombang.
- Pemeliharaan host tidak dapat diaktifkan atau dinonaktifkan untuk host yang sudah berada dalam grup sumber daya host. Host yang ditambahkan ke grup sumber daya host mempertahankan pengaturan pemeliharaan host-nya. Untuk informasi selengkapnya, lihat Grup sumber daya host.
- Pemeliharaan host tidak didukung dengan jenis instans berikut, karena mereka memiliki volume root penyimpanan instans: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

### Layanan terkait

Host Khusus terintegrasi dengan Manajer AWS Lisensi —Melacak lisensi di Amazon EC2 Dedicated Host Anda (hanya didukung di Wilayah di mana Manajer AWS Lisensi tersedia). Untuk informasi selengkapnya, lihat Panduan Pengguna AWS License Manager.

Anda harus memiliki lisensi yang cukup Akun AWS untuk Host Khusus baru Anda. Lisensi yang terkait dengan host terdegradasi dirilis saat host dirilis setelah penyelesaian acara yang dijadwalkan.

Harga

Tidak ada biaya tambahan untuk penggunaan pemeliharaan host, tetapi biaya Host Khusus yang biasa berlaku. Untuk informasi selengkapnya, lihat Harga Host Khusus Amazon EC2.

Segera setelah pemeliharaan host dimulai, Anda tidak lagi ditagih untuk Host Khusus yang terdegradasi. Tagihan untuk Host Khusus pengganti dimulai hanya setelah host masuk dalam status `available`.

Jika Host Khusus yang terdegradasi ditagih menggunakan On-Demand tarif, Host Khusus pengganti juga ditagih menggunakan tarif tersebut. On-Demand Jika Host Khusus yang terdegradasi memiliki Reservasi Host Khusus yang aktif, Host Khusus tersebut akan ditransfer ke Host Khusus yang baru.

Cara kerja pemeliharaan host untuk Amazon EC2 Dedicated Host

Ketika degradasi terdeteksi pada Host Khusus yang diaktifkan untuk pemeliharaan host, Host Khusus pengganti dialokasikan secara otomatis di akun Anda. Host Khusus pengganti menerima ID host baru, tetapi mempertahankan atribut yang sama dengan Host Khusus yang asli, termasuk:

- Pengaturan penempatan otomatis
- Zona Ketersediaan
- Asosiasi Reservasi Tuan Rumah Khusus
- Afinitas host
- Pengaturan pemeliharaan host
- Pengaturan pemulihan host
- Tipe instans
- Tanda

Setelah host pengganti dialokasikan, instans dimigrasikan menggunakan pemeliharaan host migrasi langsung  atau pemeliharaan host  berbasis reboot, tergantung instansnya.

Setelah host yang terdegradasi tidak memiliki instans yang berjalan lagi, host tersebut akan dilepaskan secara permanen dari akun Anda.

Pemeliharaan host migrasi langsung

Instans yang memerlukan pemeliharaan host migrasi langsung secara otomatis dimigrasi ke host pengganti dalam waktu 24 jam, tanpa menghentikan dan memulai ulang. Instans yang dimigrasi mempertahankan atribut yang ada, termasuk:

- ID Instans
- Metadata instans
- Lampiran volume Amazon EBS
- Alamat IP elastis dan alamat IP pribadi
- Memori, CPU, dan status jaringan

Beberapa ukuran instans yang lebih besar mungkin mengalami sedikit penurunan kinerja selama migrasi.

Setelah instans dimigrasi secara otomatis ke host pengganti, Anda menerima email dan pemberitahuan AWS Health Dasbor. Pemberitahuan mencakup ID host terdegradasi dan pengganti, informasi tentang instans yang dimigrasi secara otomatis menggunakan pemeliharaan host migrasi langsung, dan informasi tentang instans yang tersisa.

Reboot-based pemeliharaan host

Instans yang memerlukan pemeliharaan host berbasis reboot dijadwalkan misalnya acara terjadwal reboot selama 14 hari sejak tanggal pemberitahuan. Anda dapat terus mengakses instans di Host Khusus yang terdegradasi sebelum acara yang dijadwalkan.

Anda dapat menjadwal ulang acara reboot untuk tanggal yang berada dalam 7 hari dari tanggal dan waktu acara asli. Untuk informasi selengkapnya, lihat Menjadwalkan ulang acara terjadwal untuk instans EC2.

Amazon EC2 secara otomatis mencadangkan kapasitas pada host pengganti untuk instans ini. Anda tidak dapat menjalankan instance dalam kapasitas cadangan ini.

Konsol Amazon EC2 menunjukkan kapasitas cadangan sebagai kapasitas yang digunakan. Tampaknya instance berjalan pada host yang terdegradasi dan host pengganti. Namun, instance akan terus berjalan hanya pada host yang terdegradasi sampai dihentikan atau dimigrasikan ke kapasitas cadangan pada host pengganti.

Pada tanggal dan waktu acara yang dijadwalkan, instans secara otomatis dihentikan dan dimulai ulang ke kapasitas yang dicadangkan pada host pengganti. Instans yang dimigrasi mempertahankan atribut yang ada, termasuk:

- ID Instans
- Metadata instans
- Lampiran volume Amazon EBS

- Alamat IP elastis dan alamat IP pribadi

Namun, karena instance dihentikan dan dimulai ulang selama migrasi, instans tidak mempertahankan status memori, CPU, dan jaringan.

Anda juga dapat menghentikan dan memulai ulang instans ini secara manual kapan saja sebelum acara yang dijadwalkan untuk memigrasinya ke host pengganti atau ke host lain. Anda mungkin perlu mengubah afinitas host instans Anda untuk memulai ulang di host yang berbeda. Jika Anda menghentikan instance sebelum acara yang dijadwalkan, kapasitas yang dicadangkan pada host pengganti akan dirilis dan tersedia untuk digunakan.

Status pemeliharaan host

Ketika tuan rumah menjadi terdegradasi, ia memasuki `permanent-failure` negara bagian. Anda tidak dapat meluncurkan instans pada Host Khusus yang berada di `permanent-failure` negara bagian tersebut.

Setelah host pengganti dialokasikan, host tetap dalam `pending` status sampai instans yang mendukung pemeliharaan host migrasi langsung dimigrasi secara otomatis dari host yang terdegradasi, dan sampai peristiwa terjadwal dijadwalkan untuk instans yang tersisa. Setelah tugas-tugas ini selesai, tuan rumah pengganti memasuki `available` negara bagian.

Setelah host pengganti memasuki `available` status, Anda dapat menggunakannya dengan cara yang sama seperti Anda menggunakan host apa pun di akun Anda. Namun, beberapa kapasitas instans pada host pengganti dicadangkan untuk instans yang memerlukan migrasi host berbasis reboot. Anda tidak dapat meluncurkan instans baru ke kapasitas cadangan ini.

Ketika host yang terdegradasi tidak memiliki instans yang berjalan lagi, ia memasuki `released, permanent-failure` status, dan dilepaskan secara permanen dari akun Anda. Perhatikan bahwa host dan sumber dayanya tetap terlihat di konsol untuk waktu yang singkat.

Migrasi otomatis

Beberapa instans tidak dapat dimigrasikan secara otomatis ke host pengganti.

Instans dengan volume EBS-backed root

Untuk kasus ini, peristiwa penghentian instans dijadwalkan selama 28 hari sejak tanggal pemberitahuan. Pada tanggal dan waktu acara yang dijadwalkan, instance dihentikan. Sebaiknya Anda berhenti secara manual saat memulai ulang instans di host pengganti atau di host lain. Anda mungkin perlu mengubah afinitas host instans Anda untuk memulai ulang di host yang berbeda.

Instans dengan instance menyimpan volume root

Untuk kasus ini, acara pensiun misalnya dijadwalkan selama 28 hari sejak tanggal pemberitahuan. Pada tanggal dan waktu acara yang dijadwalkan, instance dihentikan secara permanen. Sebaiknya Anda meluncurkan instans pengganti secara manual di host pengganti dan kemudian memigrasikan data yang diperlukan ke instans pengganti sebelum acara yang dijadwalkan.

Instans berikut memiliki volume root penyimpanan instance: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

Anda dapat terus mengakses instans di Host Khusus yang terdegradasi sebelum acara yang dijadwalkan.

Mengonfigurasi pengaturan pemeliharaan host untuk Host Khusus Amazon EC2

Aktifkan pemeliharaan host untuk memastikan instans yang berjalan di Host Khusus secara otomatis dipulihkan ke Host Khusus baru selama acara pemeliharaan terjadwal.

Jika Anda menonaktifkan pemeliharaan host, Anda akan menerima notifikasi email untuk mengeluarkan host yang terdegradasi dan memigrasikan instans Anda secara manual ke host lain dalam waktu 28 hari. Host pengganti dialokasikan jika Anda memiliki reservasi Host Khusus. Setelah 28 hari, instans yang berjalan pada host terdegradasi akan diakhiri, dan host dilepaskan secara otomatis.

Console

Untuk mengaktifkan pemeliharaan host untuk Host Khusus Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih Host Dedikasi > Tindakan > Modifikasi host.
4. Pilih aktif di bidang Pemeliharaan host.

Untuk menonaktifkan pemeliharaan host untuk Host Khusus Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Pilih Host Dedikasi > Tindakan > Modifikasi host.
4. Pilih nonaktif di bidang Pemeliharaan host.

AWS CLI

Untuk mengaktifkan pemeliharaan host untuk Host Khusus Anda

Gunakan perintah modify-hosts.

```
aws ec2 modify-hosts \
    --host-maintenance on \
    --host-ids h-0d123456bbf78910d
```

Untuk menonaktifkan pemeliharaan host untuk Host Khusus Anda

Gunakan perintah modify-hosts.

```
aws ec2 modify-hosts \
    --host-maintenance off \
    --host-ids h-0d123456bbf78910d
```

PowerShell

Untuk mengaktifkan pemeliharaan host untuk Host Khusus Anda

Gunakan Edit-EC2Host cmdlet.

```
Edit-EC2Host `
    -HostMaintenance on `
    -HostId h-0d123456bbf78910d
```

Untuk menonaktifkan pemeliharaan host untuk Host Khusus Anda

Gunakan Edit-EC2Host cmdlet.

```
Edit-EC2Host `
    -HostMaintenance off `
    -HostId h-0d123456bbf78910d
```

## Pantau status Host Khusus Amazon EC2 Anda

Amazon EC2 secara konstan memantau status Host Khusus Anda. Pembaruan dikomunikasikan di konsol Amazon EC2. Anda dapat melihat informasi tentang Host Khusus menggunakan metode berikut ini.

Console

Untuk melihat status Host Khusus

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Host Khusus.
3. Cari Host khusus dalam daftar dan tinjau nilai di kolom Status.

AWS CLI

Untuk melihat status Host Khusus

Gunakan perintah describe-host.

```
aws ec2 describe-hosts --host-id h-012a3456b7890cdef
```

PowerShell

Untuk melihat status Host Khusus

Gunakan Get-EC2Host cmdlet.

```
Get-EC2Host -HostId h-012a3456b7890cdef
```

Tabel berikut menjelaskan kemungkinan status Host Khusus.

| Status | Deskripsi |
| --- | --- |
| `available` | AWS belum mendeteksi masalah dengan Host Khusus. Tidak ada pemeliharaan atau perbaikan yang dijadwalkan. Instans dapat diluncurkan ke Host Khusus ini. |
| `released` | Host Khusus telah dilepas. ID host tidak lagi digunakan. Host yang dilepas tidak dapat digunakan kembali. |
| `under-assessment` | AWS sedang mengeksplorasi kemungkinan masalah dengan Host Khusus. Jika tindakan harus diambil, Anda diberitahu melalui Konsol Manajemen AWS atau email. Instans tidak dapat diluncurkan ke Host Khusus dalam status ini. |

| Status | Deskripsi |
| --- | --- |
| `pending` | Host Khusus tidak dapat digunakan untuk peluncuran instans baru. Modifikasi untuk mendukung banyak tipe instans atau pemulihan host sedang berlangsung. |
| `permanent-failure` | Telah terdeteksi kegagalan yang tidak dapat dipulihkan. Anda menerima pemberitahuan pengosongan melalui instans Anda dan melalui email. Instans Anda mungkin terus berjalan. Jika Anda menghentikan atau menghentikan semua instans di Host Khusus dengan status ini, host AWS akan pensiun. AWS tidak memulai ulang instance dalam status ini. Instans tidak dapat diluncurkan ke Host Khusus dalam status ini. |
| `released-permanent-failure` | AWS merilis Host Khusus secara permanen yang gagal dan tidak lagi memiliki instans yang berjalan di dalamnya. ID Host Khusus tidak lagi tersedia untuk digunakan. |

## Lacak perubahan konfigurasi Host Khusus Amazon EC2 menggunakan AWS Config

Anda dapat menggunakannya AWS Config untuk merekam perubahan konfigurasi untuk Host Khusus, dan untuk instans yang diluncurkan, dihentikan, atau dihentikan pada host tersebut. Anda kemudian dapat menggunakan informasi yang ditangkap oleh AWS Config sebagai sumber data untuk pelaporan lisensi.

AWS Config merekam informasi konfigurasi untuk Host Khusus dan instans secara individual, dan memasangkan informasi ini melalui hubungan. Ada tiga kondisi pelaporan:

- AWS Config status perekaman —Saat Aktif, AWS Config merekam satu atau lebih jenis AWS sumber daya, yang dapat mencakup Host Khusus dan Instans Khusus. Untuk menangkap informasi yang diperlukan untuk pelaporan lisensi, pastikan bahwa host dan instans sedang direkam dengan bidang berikut.
- Status pencatatan host—Saat Diaktifkan, informasi konfigurasi untuk Host Khusus dicatat.
- Status pencatatan instans—Saat Diaktifkan, informasi konfigurasi untuk Instans Khusus dicatat.

Jika salah satu dari ketiga kondisi ini dinonaktifkan, ikon di tombol Edit Config Recording akan berwarna merah. Untuk mendapatkan manfaat penuh dari alat ini, pastikan bahwa ketiga metode pencatatan diaktifkan. Jika ketiganya diaktifkan, ikon akan berwarna hijau. Untuk mengedit

pengaturan, pilih Edit Pencatatan Konfigurasi. Anda diarahkan ke AWS Config halaman Pengaturan di AWS Config konsol, tempat Anda dapat mengatur AWS Config dan mulai merekam untuk host, instans, dan jenis sumber daya lain yang didukung. Untuk informasi selengkapnya, lihat Men yi AWS Config apkan menggunakan Konsol di Panduan AWS Config Pengembang.

> **Note**
>
> AWS Config mencatat sumber daya Anda setelah menemukannya, yang mungkin memakan waktu beberapa menit.

Setelah AWS Config mulai merekam perubahan konfigurasi pada host dan instans, Anda bisa mendapatkan riwayat konfigurasi host apa pun yang telah dialokasikan atau dirilis dan instans apa pun yang telah diluncurkan, dihentikan, atau dihentikan. Misalnya, di titik mana pun dalam riwayat konfigurasi Host Khusus, Anda dapat mencari berapa banyak instans yang diluncurkan pada host tersebut, bersama dengan jumlah soket dan inti pada host. Untuk salah satu instans tersebut, Anda juga dapat mencari ID dari Amazon Machine Image (AMI). Anda dapat menggunakan informasi ini untuk melaporkan tentang pelisensian untuk perangkat lunak terikat server Anda sendiri yang mendapat lisensi per soket atau per inti.

Anda dapat melihat riwayat konfigurasi dengan salah satu dari cara berikut ini:

- Dengan menggunakan AWS Config konsol. Untuk setiap sumber daya yang tercatat, Anda dapat melihat halaman kronologi, yang memberikan detail riwayat konfigurasi. Untuk melihat halaman ini, pilih ikon abu-abu di kolom Konfigurasi Kronologi pada halaman Host Khusus. Untuk informasi selengkapnya, lihat Melihat Detail Konfigurasi di AWS Config Konsol di Panduan AWS Config Pengembang.
- Dengan menjalankan AWS CLI perintah. Pertama, Anda dapat menggunakan perintah list-found-resources untuk mendapatkan daftar semua host dan instans. Kemudian, Anda dapat menggunakan perintah get-resource-config-history untuk mendapatkan detail konfigurasi host atau instans untuk interval waktu tertentu.
- Dengan menggunakan AWS Config API dalam aplikasi Anda. Pertama, Anda dapat menggunakanListDiscoveredResources tindakan untuk mendapatkan daftar semua host dan instance. Kemudian, Anda dapat menggunakanGetResourceConfigHistory tindakan untuk mendapatkan detail konfigurasi host atau instance untuk interval waktu tertentu.

Misalnya, untuk mendapatkan daftar semua Host Khusus Anda dari AWS Config, jalankan perintah CLI seperti berikut ini.

```
aws configservice list-discovered-resources --resource-type AWS::EC2::Host
```

Untuk mendapatkan riwayat konfigurasi Host Khusus dari AWS Config, jalankan perintah CLI seperti berikut ini.

```
aws configservice get-resource-config-history \
    --resource-type AWS::EC2::Instance \
    --resource-id i-1234567890abcdef0
```

Untuk mengelola AWS Config pengaturan menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di halaman Host Khusus, pilih Edit Pencatatan Konfigurasi.
3. Di AWS Config konsol, ikuti langkah-langkah yang disediakan untuk mengaktifkan perekaman. Untuk informasi selengkapnya, lihat Menyi AWS Config apkan menggunakan Konsol.

Untuk informasi selengkapnya, lihat Melihat Detail Konfigurasi di AWS Config Konsol.

Untuk mengaktifkan AWS Config menggunakan baris perintah atau API

- AWS CLI: Melihat Detail Konfigurasi (AWS CLI) di Panduan Peng AWS Config embang.
- API Amazon EC2: GetResourceConfigHistory.

## Instans Khusus Amazon EC2

Secara default, instans EC2 berjalan pada perangkat keras penghunian bersama. Ini berarti bahwa beberapa AWS akun mungkin berbagi perangkat keras fisik yang sama.

Instans Khusus adalah instans EC2 yang berjalan pada perangkat keras yang didedikasikan untuk satu AWS akun. Ini berarti bahwa Instans Khusus secara fisik terisolasi di tingkat perangkat keras host dari instans milik lain Akun AWS, bahkan jika akun tersebut ditautkan ke akun pembayar tunggal. Namun, Instans Khusus mungkin berbagi perangkat keras dengan instans lain dari instans yang sama Akun AWS yang bukan Instans Khusus.

Instans Khusus tidak menyediakan visibilitas atau kontrol atas penempatan instans, dan tidak mendukung afinitas host. Jika Anda menghentikan dan memulai Dedicated Instance, itu mungkin tidak berjalan di host yang sama. Demikian pula, Anda tidak dapat menargetkan host tertentu untuk meluncurkan atau menjalankan instance. Selain itu, Instans Khusus menyediakan dukungan terbatas untuk Bring Your Own License (BYOL).

Jika Anda memerlukan visibilitas dan kontrol atas penempatan instans dan dukungan BYOL yang lebih komprehensif, pertimbangkan untuk menggunakan Host Khusus sebagai gantinya. Instans Khusus dan Host Khusus keduanya dapat digunakan untuk meluncurkan instans Amazon EC2 ke server fisik khusus. Tidak ada perbedaan performa, keamanan, atau fisik di antara Instans Khusus dan instans pada Host Khusus. Namun, ada beberapa perbedaan utama di antara mereka. Tabel berikut menyoroti beberapa perbedaan utama antara Instans Khusus dan Host Khusus:

| | Host Khusus | Instans Khusus |
|---|---|---|
| Server fisik khusus | Server fisik dengan kapasitas instance yang sepenuhnya didedikasikan untuk penggunaan Anda. | Server fisik yang didedikasikan untuk satu akun pelanggan. |
| Berbagi kapasitas instans | Dapat berbagi kapasitas instance dengan akun lain. | Tidak didukung |
| Penagihan | Per-host penagihan | Per-instance penagihan |
| Visibilitas soket, inti, dan ID host | Memberikan visibilitas dalam jumlah soket dan inti fisik | Tidak ada visibilitas |
| Afinitas host dan instans | Memungkinkan Anda melakukan deployment instans Anda secara konsisten ke server fisik yang sama seiring waktu | Tidak didukung |
| Penempatan instans tertarget | Memberikan visibilitas dan kontrol tambahan atas cara penempatan instans di server fisik | Tidak didukung |

| | Host Khusus | Instans Khusus |
|---|---|---|
| Pemulihan instans otomatis | Didukung. Untuk informasi selengkapnya, lihat Pemulihan Host Khusus Amazon EC2. | Didukung |
| Bawa Lisensi Sendiri (BYOL) | Didukung | Dukungan parsial* |
| Reservasi Kapasitas | Tidak didukung | Didukung |

* Microsoft SQL Server dengan License Mobility melalui Jaminan Perangkat Lunak, sedangkan lisensi Windows Virtual Desktop Access (VDA) dapat digunakan dengan Instans Khusus.

Untuk informasi selengkapnya, lihat Host Khusus Amazon EC.

Daftar Isi


- Dasar-dasar Instans Khusus
- Fitur yang didukung
- Batasan Instans Khusus
- Harga untuk Instans Khusus
- Luncurkan Instans Khusus ke VPC dengan sewa default
- Mengubah sewa instans EC2
- Mengubah sewa instans VPC


## Dasar-dasar Instans Khusus

VPC dapat memiliki penghunian `default` atau `dedicated`. Secara default, VPC Anda memiliki penghunian `default` dan instans yang diluncurkan ke VPV penghunian `default` memiliki penghunian `default`. Untuk meluncurkan Instans Khusus, lakukan hal berikut:

- Buat VPC dengan penghunian `dedicated`, sehingga semua instans di VPC berjalan sebagai Instans Khusus. Untuk informasi selengkapnya, lihat Luncurkan Instans Khusus ke VPC dengan sewa default.

- Buat VPC dengan penghunian `default` dan tentukan secara manual penghunian `dedicated` untuk instans agar dijalankan sebagai Instans Khusus. Untuk informasi selengkapnya, lihat Luncurkan Instans Khusus ke VPC dengan sewa default.

## Fitur yang didukung

Instans Khusus mendukung fitur dan integrasi AWS layanan berikut:

Fitur

- Instans Terpesan
- Penskalaan Otomatis
- Pemulihan otomatis
- Instans Spot Khusus
- Instance performa yang dapat melonjak

Instans Terpesan

Untuk mencadangkan kapasitas Instans Khusus, Anda dapat membeli Instans Cadangan Khusus atau Reservasi Kapasitas. Untuk informasi selengkapnya, lihat Ikhtisar Instans Cadangan untuk Amazon EC2 dan Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand.

Saat membeli Instans Cadangan Khusus, Anda membeli kapasitas untuk meluncurkan Instans Khusus dengan biaya penggunaan yang jauh lebih murah; potongan harga dalam biaya penggunaan hanya berlaku jika Anda meluncurkan instans dengan sewa khusus. Saat Anda membeli Instans Terpesan dengan penghunian default, hal ini hanya berlaku untuk instans yang sedang berjalan dengan penghunian `default`, tetapi tidak berlaku untuk instans yang sedang berjalan dengan penghunian `dedicated`.

Anda tidak dapat menggunakan proses modifikasi untuk mengubah penghunian Instans Terpesan setelah Anda membelinya. Namun, Anda dapat menukar Instans Terpesan Konvertibel dengan Instans Terpesan Konvertibel baru dengan penghunian yang berbeda.

Penskalaan Otomatis

Anda dapat menggunakan Amazon EC2 Auto Scaling untuk meluncurkan Instans Khusus. Untuk informasi selengkapnya, lihat Membuat template peluncuran menggunakan pengaturan lanjutan di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

Pemulihan otomatis

Anda dapat mengonfigurasi pemulihan otomatis untuk Instans Khusus jika menjadi terganggu karena kegagalan perangkat keras yang mendasarinya atau masalah yang memerlukan AWS keterlibatan untuk diperbaiki. Untuk informasi selengkapnya, lihat Pemulihan instans otomatis.

Instans Spot Khusus

Anda dapat menjalankan Instans Spot Khusus dengan menentukan penghunian `dedicated` saat Anda membuat permintaan Instans Spot. Untuk informasi selengkapnya, lihat Peluncuran pada perangkat keras penyewa tunggal.

Instance performa yang dapat melonjak

Anda dapat memanfaatkan keuntungan menjalankan perangkat keras penghunian khusus dengan the section called “Instance performa yang dapat melonjak”. Instans Khusus T3 diluncurkan dalam mode tidak terbatas secara default, dan mereka memberikan tingkat acuan performa CPU dengan kemampuan untuk melonjak ke tingkat CPU yang lebih tinggi saat diperlukan oleh beban kerja Anda. Performa acuan T3 dan kemampuan melonjak diatur oleh kredit CPU. Karena sifat tipe instans T3 yang dapat melonjak, kami menyarankan Anda untuk memantau bagaimana instans T3 Anda menggunakan sumber daya CPU dari perangkat keras khusus untuk mendapatkan performa terbaik. Instans Khusus T3 ditujukan untuk pelanggan dengan beban kerja beragam yang menampilkan perilaku CPU acak, tetapi idealnya memiliki penggunaan CPU rata-rata pada atau di bawah penggunaan acuan. Untuk informasi selengkapnya, lihat the section called “Konsep Utama”.

Amazon EC2 memiliki sistem untuk mengidentifikasi dan mengoreksi variabilitas dalam performa. Namun, masih mungkin untuk mengalami variabilitas jangka pendek jika Anda meluncurkan beberapa Instans Khusus T3 yang memiliki pola penggunaan CPU yang berkorelasi. Untuk beban kerja yang lebih menuntut atau berkorelasi ini, kami merekomendasikan penggunaan Instans Khusus M5 atau M5a daripada Instans Khusus T3.

## Batasan Instans Khusus

Ingatlah hal-hal berikut ini saat menggunakan Instans Khusus:

- Beberapa AWS layanan atau fiturnya tidak didukung dengan VPC dengan penyewaan instans disetel ke. `dedicated` Lihat dokumentasi layanan masing-masing untuk mengonfirmasi jika ada batasan.

- Beberapa tipe instans tidak dapat diluncurkan ke VPC dengan penghunian instans yang diatur ke `dedicated`. Untuk informasi selengkapnya tentang tipe instans yang didukung, lihat Instans Khusus Amazon EC2.
- Saat Anda meluncurkan Instans Khusus yang didukung Amazon EBS, volume EBS tidak berjalan pada perangkat keras penghunian tunggal.

## Harga untuk Instans Khusus

Harga untuk Instans Khusus berbeda dengan harga untuk On-Demand Instans. Untuk informasi selengkapnya, lihat Inst ans Khusus Amazon EC2.

## Luncurkan Instans Khusus ke VPC dengan sewa default

Saat membuat VPC, Anda memiliki opsi untuk menentukan penghunian instansnya. Jika Anda meluncurkan instance ke VPC yang memiliki sewa instans`dedicated`, instans tersebut akan berjalan sebagai Instans Khusus pada perangkat keras yang didedikasikan untuk Anda gunakan.

Untuk informasi selengkapnya tentang meluncurkan instans dengan penghunian `host`, lihat Luncurkan instans Amazon EC2 di Host Khusus Amazon EC2.

Untuk informasi selengkapnya tentang opsi sewa VPC, lihat Membuat VPC di Panduan Pengguna Amazon VPC.

Persyaratan

- Pilih jenis instans yang didukung. Untuk informasi selengkapnya, lihat Amazon EC2 Dedicated instans.

Console

Untuk meluncurkan Instans Khusus ke VPC sewa default

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, Luncurkan instans.
3. Di bagian Gambar Aplikasi dan OS, pilih AMI dari daftar.
4. Di bagian tipe instans, pilih tipe instans yang akan diluncurkan.
5. Di bagian Key pair, pilih key pair untuk diasosiasikan dengan instans.
6. Di bagian Detail lanjutan, untuk Penghunian, pilih Khusus.

7. Konfigurasikan opsi instans yang tersisa sesuai kebutuhan. Untuk informasi selengkapnya, lihat Referensi untuk parameter konfigurasi EC2 instans Amazon.
8. Pilih Luncurkan instans.

AWS CLI

Untuk mengatur opsi sewa untuk instance selama peluncuran

Gunakan perintah run-instance dan sertakan `Tenancy` dengan opsi. `--placement`

```
--placement Tenancy=dedicated
```

PowerShell

Untuk mengatur opsi sewa untuk instance selama peluncuran

Gunakan New-EC2Instance cmdlet dengan parameter. `-Placement_Tenancy`

```
-Placement_Tenancy dedicated
```

## Mengubah sewa instans EC2

Anda dapat mengubah penghunian instans yang dihentikan setelah peluncuran. Perubahan yang Anda buat berlaku saat berikutnya instans dimulai.

Atau, Anda dapat mengubah sewa cloud pribadi virtual (VPC) Anda. Untuk informasi selengkapnya, lihat the section called “Mengubah penghunian VPC”.

Batasan

- Anda tidak dapat mengubah sewa instance menggunakan. Konsol Manajemen AWS
- Instans harus berada dalam status `stopped`.
- Detail sistem operasi instans Anda, dan apakah SQL Server diinstal, memengaruhi konversi apa yang didukung. Untuk informasi selengkapnya tentang jalur konversi penghunian yang tersedia untuk instans Anda, lihat Konversi penghunia di Panduan Pengguna Manajer Lisensi.
- Untuk instans T3, Anda harus meluncurkan instans pada Host Khusus untuk menggunakan penyewaan `host`. Anda tidak dapat mengubah penghunian dari `host` menjadi `dedicated` atau `default`. Percobaan mengubah salah satu penghunian yang tidak didukung ini dapat mengakibatkan kode kesalahan `InvalidRequest`.

AWS CLI

Untuk memodifikasi nilai sewa sebuah instance

Gunakan perintah modify- instance-placement.

```
aws ec2 modify-instance-placement \
    --instance-id i-1234567890abcdef0 \
    --tenancy dedicated
```

PowerShell

Untuk memodifikasi nilai sewa sebuah instance

Gunakan Edit-EC2InstancePlacement cmdlet.

```
Edit-EC2InstancePlacement `
    -InstanceId i-1234567890abcdef0 `
    -Tenancy Dedicated
```

## Mengubah sewa instans VPC

Anda dapat mengubah sewa instans cloud pribadi virtual (VPC) dari `dedicated` menjadi `default` setelah Anda membuatnya. Memodifikasi sewa instans VPC tidak mempengaruhi sewa instans yang ada di VPC. Lain kali Anda meluncurkan instance di VPC, instans tersebut memiliki sewa`default`, kecuali jika Anda menentukan sebaliknya selama peluncuran instance.

Atau, Anda dapat mengubah sewa instans tertentu. Untuk informasi selengkapnya, lihat the section called “Mengubah penghunian suatu instans”.

Batasan

- Anda tidak dapat mengubah sewa instans VPC dari `default` ke `dedicated` setelah dibuat.
- Anda tidak dapat mengubah sewa instans VPC menggunakan. Konsol Manajemen AWS

AWS CLI

Untuk memodifikasi atribut penyewaan instans dari VPC

Gunakan perintah modify- vpc-tenancy. Satu-satunya nilai sewa yang didukung adalah`default`.

```
aws ec2 modify-vpc-tenancy \
    --vpc-id vpc-1234567890abcdef0 \
    --instance-tenancy default
```

PowerShell

Untuk memodifikasi atribut penyewaan instans dari VPC

Gunakan Edit-EC2VpcTenancy cmdlet. Satu-satunya nilai sewa yang didukung adalah`Default`.

```
Edit-EC2VpcTenancy `
    -VpcId vpc-1234567890abcdef0 `
    -InstanceTenancy Default
```

## On-Demand Reservasi Kapasitas dan Blok Kapasitas untuk ML

Reservasi Kapasitas memungkinkan Anda memesan kapasitas komputasi untuk instans Amazon EC2 di Zona Ketersediaan tertentu. Ada dua tipe Reservasi Kapasitas yang melayani kasus penggunaan yang berbeda.

Tipe Reservasi Kapasitas

- On-Demand Reservasi Kapasitas
- Blok Kapasitas untuk ML

Untuk mengatur Reservasi Kapasitas ke dalam kumpulan logis, lihat Grup Sumber Daya Reservasi Kapasitas.

Berikut ini adalah beberapa kasus penggunaan umum untuk Reservasi On-Demand Kapasitas:

- Peristiwa penskalaan — Buat Reservasi On-Demand Kapasitas sebelum acara penting bisnis Anda untuk memastikan bahwa Anda dapat menskalakan kapan pun Anda perlu.
- Persyaratan peraturan dan pemulihan bencana — Gunakan On-Demand Reservasi Kapasitas untuk memenuhi persyaratan peraturan untuk ketersediaan tinggi, dan cadangan kapasitas di Zona Ketersediaan atau Wilayah yang berbeda untuk pemulihan bencana.
- Berbagi kapasitas yang tidak terpakai — Gunakan Reservasi Kapasitas Terputus untuk membuat kapasitas yang tidak terpakai tersedia untuk beban kerja lain dalam akun Anda sambil mempertahankan kontrol untuk merebutnya kembali saat diperlukan.

Berikut ini adalah beberapa kasus penggunaan umum untuk Blok Kapasitas untuk ML:

- Pelatihan model machine learning (ML) dan fine-tuning — Dapatkan akses tanpa gangguan ke instans GPU yang Anda pesan untuk menyelesaikan pelatihan model dan fine-tuning.
- Eksperimen dan prototipe ML — Jalankan eksperimen dan bangun prototipe yang memerlukan instans GPU untuk jangka waktu pendek.

Kapan menggunakan Reservasi On-Demand Kapasitas

Gunakan Reservasi On-Demand Kapasitas jika Anda memiliki persyaratan kapasitas yang ketat, dan beban kerja kritis bisnis Anda saat ini atau masa depan memerlukan jaminan kapasitas. Dengan On-Demand Reservasi Kapasitas, Anda dapat memastikan bahwa Anda akan selalu memiliki akses ke kapasitas Amazon EC2 yang telah Anda pesan selama Anda membutuhkannya.

Kapan menggunakan Blok Kapasitas untuk ML

Gunakan Blok Kapasitas untuk ML saat Anda perlu memastikan bahwa Anda memiliki akses tanpa gangguan ke instans GPU untuk jangka waktu tertentu yang dimulai pada tanggal yang akan datang. Blok Kapasitas ideal untuk melatih dan menyempurnakan model ML, menjalankan eksperimen singkat, dan menangani lonjakan sementara dalam permintaan inferensi di masa mendatang. Dengan Blok Kapasitas, Anda dapat memastikan bahwa Anda akan memiliki akses ke sumber daya GPU pada tanggal tertentu untuk menjalankan beban kerja ML Anda.

## Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand

Reservasi Kapasitas Amazon EC2 memungkinkan Anda untuk memesan kapasitas komputasi untuk instans Amazon EC2 Anda di Zona Ketersediaan tertentu untuk durasi apa pun. Jika Anda memiliki persyaratan kapasitas yang ketat untuk beban kerja kritis bisnis saat ini atau masa depan yang memerlukan tingkat jaminan kapasitas jangka panjang atau jangka pendek tertentu, kami sarankan Anda membuat Reservasi Kapasitas untuk membantu memastikan bahwa Anda selalu memiliki akses ke kapasitas Amazon EC2 saat Anda membutuhkannya, selama Anda membutuhkannya.

Anda dapat membuat Reservasi Kapasitas kapan saja, dan Anda dapat memilih kapan reservasi itu dimulai. Anda dapat meminta Reservasi Kapasitas untuk penggunaan segera atau Anda dapat meminta Reservasi Kapasitas untuk tanggal mendatang.

- Jika Anda meminta Reservasi  Kapasitas untuk penggunaan segera, Reservasi Kapasitas akan segera tersedia untuk digunakan dan tidak ada komitmen jangka panjang. Anda dapat mengubah

Reservasi Kapasitas kapan saja, dan Anda dapat membatalkannya kapan saja untuk melepaskan kapasitas yang dipesan dan untuk menghentikan biaya yang dikenakan.

- Jika Anda meminta Reservasi  Kapasitas di masa mendatang, Anda menentukan kapan Anda membutuhkan kapasitas dan berapa lama Anda berkomitmen untuk menyimpannya. Pada tanggal yang ditentukan di masa mendatang, Reservasi Kapasitas tersedia untuk digunakan dan penagihan dimulai. Selama durasi komitmen, Anda tidak dapat mengurangi jumlah instans atau durasi komitmen di bawah komitmen awal Anda. Anda dapat membatalkan Reservasi Kapasitas, tetapi biaya pembatalan mungkin berlaku tergantung kapan Anda membatalkan. Setelah durasi komitmen berlalu, Anda dapat mengubah Reservasi Kapasitas dengan cara apa pun atau membatalkannya tanpa biaya.

Reservasi Kapasitas hanya dapat digunakan oleh instans yang cocok dengan atributnya. Secara default, Reservasi Kapasitas secara otomatis mencocokkan instans baru dan instans yang berjalan yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa). Ini berarti bahwa setiap instans dengan atribut yang cocok secara otomatis berjalan di Reservasi Kapasitas. Namun, Anda juga dapat menargetkan Reservasi Kapasitas untuk beban kerja tertentu. Ini memungkinkan Anda untuk secara eksplisit mengontrol instans mana yang diizinkan untuk dijalankan dalam kapasitas yang dicadangkan tersebut. Anda juga dapat menentukan bahwa instance hanya akan berjalan di Grup Sumber Daya Reservasi Kapasitas atau Reservasi Kapasitas.

> **Important**
>
> Future-dated Reservasi Kapasitas adalah untuk membantu Anda meluncurkan dan mencakup instans inkremental, dan bukan untuk mencakup instans yang sedang berjalan. Jika Anda perlu mencakup instans yang sedang berjalan, gunakan Reservasi Kapasitas yang segera dimulai sebagai gantinya.

Semua instans Amazon EC2 yang didukung dengan atribut yang cocok, yaitu jenis instans, platform, Zona Ketersediaan, dan sewa, memenuhi syarat untuk dijalankan dalam Reservasi Kapasitas. Instans Amazon EC2 dapat diluncurkan oleh Anda (instans yang  tidak dikelola) atau atas nama Anda oleh AWS layanan (inst  ans terkel  ola). Hal ini terutama berlaku untuk Reservasi  Kapasitas terbuka, yang secara otomatis cocok dengan instans yang sedang berjalan yang memiliki atribut yang cocok. Misalnya, instans terkelola yang diluncurkan atas nama Anda oleh layanan berikut memenuhi syarat untuk dijalankan di Reservasi Kapasitas yang Anda buat dan kelola.

- Amazon EC2 Auto Scaling

- Amazon ECS
- Amazon EKS
- Amazon EMR
- AWS Batch
- AWS Elastic Beanstalk
- AWS ParallelCluster
- AWS Parallel Computing Service (AWS PCS)

Daftar Isi


- Konsep untuk Reservasi Kapasitas Amazon EC2
- Perbedaan antara Reservasi Kapasitas, Instans Terpesan, dan Savings Plans
- Platform yang didukung
- Kuota
- Batasan
- Harga dan penagihan Reservasi Kapasitas
- Membuat Reservasi Kapasitas
- Melihat status Reservasi Kapasitas
- Luncurkan instans ke dalam Reservasi Kapasitas yang ada
- Mengubah Reservasi Kapasitas yang aktif
- Mengubah setelan Reservasi Kapasitas instans Anda
- Memindahkan kapasitas antara Reservasi Kapasitas
- Memisahkan kapasitas dari Reservasi Kapasitas yang ada
- Membatalkan Reservasi Kapasitas
- Gunakan Reservasi Kapasitas dengan grup penempatan
- Reservasi Kapasitas di Local Zones
- Reservasi Kapasitas di Wavelength Zones
- Pencadangan Kapasitas di AWS Outposts
- Reservasi Kapasitas Bersama
- Armada Reservasi Kapasitas
- Memantau penggunaan Reservasi Kapasitas dengan CloudWatch metrik

- Memantau Pemanfaatan Pemesanan Kapasitas yang kurang
- Pantau perubahan status untuk Reservasi Kapasitas di masa mendatang
- Reservasi Kapasitas yang Dapat Diinterupsi

## Konsep untuk Reservasi Kapasitas Amazon EC2

Konsep kunci berikut berlaku untuk Reservasi Kapasitas.

Topik

- Tanggal dan waktu mulai
- Tanggal dan waktu akhir
- Durasi komitmen
- Future-dated Penilaian Reservasi Kapasitas
- Atribut Reservasi Kapasitas
- Kriteria pencocokan instans

### Tanggal dan waktu mulai

Tanggal dan waktu mulai menentukan kapan Reservasi Kapasitas tersedia untuk digunakan. Reservasi Kapasitas dapat dimulai segera, atau dapat dimulai di masa mendatang.

- Jika Anda memilih untuk segera memulai Reservasi Kapasitas, kapasitas yang dipesan akan tersedia untuk digunakan segera setelah Anda membuatnya, dan penagihan dimulai segera setelah Reservasi Kapasitas memasuki status aktif. Anda tidak perlu masuk ke dalam komitmen jangka apa pun. Anda dapat mengubah Reservasi Kapasitas sesuai kebutuhan kapan saja untuk memenuhi kebutuhan Anda, dan Anda dapat membatalkannya kapan saja untuk melepaskan kapasitas dan menghentikan biaya yang dikenakan.
- Jika Anda memilih untuk memulai Reservasi Kapasitas di masa mendatang, Anda menentukan kapan Anda membutuhkan kapasitas dan berapa lama Anda berkomitmen untuk menyimpannya. Pada tanggal mendatang yang ditentukan, Reservasi Kapasitas tersedia untuk digunakan dan penagihan dimulai setelah Reservasi Kapasitas memasuki status aktif. Durasi komitmen dimulai segera setelah Reservasi Kapasitas disediakan di akun Anda. Selama waktu ini, Anda tidak dapat mengurangi jumlah instans di bawah jumlah instans yang dikomit, atau memilih tanggal akhir sebelum durasi komitmen. Anda dapat membatalkan Reservasi Kapasitas, tetapi biaya pembatalan mungkin berlaku tergantung kapan Anda membatalkan. Untuk informasi selengkapnya, lihat

Membatalkan Reservasi Kapasitas. Setelah durasi komitmen berakhir, Anda dapat mengubah Reservasi Kapasitas dengan cara apa pun, atau membatalkannya untuk melepaskan kapasitas yang dipesan dan untuk menghentikan biaya yang dikenakan.

Tanggal dan waktu akhir

Tanggal dan waktu akhir menentukan kapan Reservasi Kapasitas berakhir dan kapasitas yang dipesan dilepaskan dari akun Anda. Anda dapat mengonfigurasi Reservasi Kapasitas agar berakhir secara otomatis pada tanggal dan waktu tertentu, atau Anda dapat mengonfigurasinya agar tetap aktif tanpa batas waktu hingga Anda membat alkannya secara manual.

Jika Anda mengonfigurasi Reservasi Kapasitas untuk diakhiri secara otomatis, Reservasi Kapasitas berakhir dalam waktu satu jam dari waktu yang ditentukan. Misalnya, jika Anda menentukan`5/31/2019, 13:30:55`, Reservasi Kapasitas dijamin akan berakhir antara `13:30:55` dan seter `14:30:55` usnya`5/31/2019`.

Setelah reservasi berakhir, kapasitas yang dipesan dilepaskan dari akun Anda dan Anda tidak dapat lagi menargetkan instans ke Reservasi Kapasitas. Instans yang berjalan dalam kapasitas terpesan terus berjalan tanpa interupsi. Jika instans yang menargetkan Reservasi Kapasitas dihentikan, Anda tidak dapat memulai ulang instans tersebut sampai Anda menghapus preferensi penargetan Reservasi Kapasitas atau mengonfigurasinya untuk menargetkan Reservasi Kapasitas yang berbeda. Untuk informasi selengkapnya, lihat Mengubah setelan Reservasi Kapasitas instans Anda.

Durasi komitmen

Durasi komitmen hanya berlaku untuk Reservasi Kapasitas di masa mendatang.

Durasi komitmen adalah durasi minimum di mana Anda berkomitmen untuk memiliki Reservasi Kapasitas di masa mendatang dalam status aktif di akun Anda setelah disediakan. Anda dapat menyimpan Reservasi Kapasitas di masa mendatang lebih lama dari durasi komitmen, tetapi tidak lebih pendek. Berikut ini berlaku selama durasi komitmen:

- Anda dapat membatalkan Reservasi Kapasitas selama durasi komitmen, tetapi biaya pembatalan mungkin berlaku. Untuk informasi selengkapnya, lihat Biaya pembatalan.
- Anda tidak dapat mengurangi jumlah instans di bawah jumlah instance yang dikomit, tetapi Anda dapat meningkatkannya.
- Anda tidak dapat mengonfigurasi Reservasi Kapasitas untuk secara otomatis berakhir pada tanggal atau waktu yang berada dalam durasi komitmen. Anda dapat memperpanjang tanggal dan waktu akhir selama durasi komitmen.

Amazon EC2 menggunakan durasi komitmen yang Anda tentukan untuk menilai apakah permintaan dapat didukung. Durasi komitmen minimum adalah 14 hari untuk sebagian besar jenis instans, atau 12 minggu untuk instans U dan X. Saat menilai permintaan, Amazon EC2 mungkin menentukan bahwa itu dapat mendukung durasi komitmen yang lebih pendek. Dalam hal ini, Amazon EC2 menjadwalkan Reservasi Kapasitas di masa mendatang dengan durasi komitmen yang lebih pendek. Ini berarti bahwa Anda berkomitmen untuk menyimpan Reservasi Kapasitas di akun Anda untuk jangka waktu yang lebih pendek dari yang Anda minta sebelumnya.

Future-dated Penilaian Reservasi Kapasitas

Saat Anda meminta Reservasi Kapasitas di masa mendatang, Amazon EC2 menilai permintaan tersebut untuk menentukan apakah permintaan tersebut dapat didukung berdasarkan ketersediaan kapasitas dan durasi komitmen yang Anda tentukan. Penilaian biasanya diselesaikan dalam waktu 5 hari. Amazon EC2 mempertimbangkan beberapa faktor saat mengevaluasi permintaan, termasuk:

- Pasokan kapasitas yang diperkirakan
- Durasi komitmen
- Seberapa awal Anda meminta Reservasi Kapasitas relatif terhadap tanggal mulai
- Ukuran permintaan Anda

Anda dapat meminta Reservasi Kapasitas di masa mendatang antara 5 dan 120 hari sebelumnya. Kami menyarankan Anda membuat permintaan setidaknya 56 hari (8 minggu) sebelumnya untuk meningkatkan kemampuan kami untuk mendukung permintaan Anda. Durasi komitmen minimum adalah 14 hari untuk sebagian besar jenis instans, atau 12 minggu untuk instans U dan X, dan jumlah instans minimum adalah 32 vCPU.

Reservasi Kapasitas tetap di `assessing` negara bagian saat permintaan sedang dinilai.

Jika permintaan dapat didukung, Reservasi Kapasitas memasuki `scheduled` negara bagian dan dijadwalkan untuk pengiriman pada tanggal dan waktu yang diminta. Jumlah instans total tetap 0 saat Reservasi Kapasitas berada dalam `scheduled` status. Reservasi Kapasitas terjadwal akan menjadi `active` dan tersedia untuk digunakan pada tanggal yang diminta.

Jika permintaan tidak dapat didukung, Reservasi Kapasitas memasuki `unsupported` status. Reservasi Kapasitas yang tidak didukung tidak dikirimkan.

Anda dapat membatalkan Reservasi Kapasitas yang akan datang di negara bagian`assessing`,`scheduled`, atau`active`. Biaya pembatalan mungkin berlaku. Untuk informasi selengkapnya, lihat Membatalkan Reservasi Kapasitas.

Untuk membuat Reservasi Kapasitas yang akan datang, lihat. Membuat Reservasi Kapasitas yang akan datang

Atribut Reservasi Kapasitas

Saat membuat Reservasi Kapasitas, Anda harus menentukan atribut berikut:

- Zona Ketersediaan
- Tipe instans
- Platform (jenis sistem operasi)
- Sewa (`defaultataudedicated`)

Hanya instance yang cocok dengan atribut ini yang dapat diluncurkan atau dijalankan di Reservasi Kapasitas.

Kriteria pencocokan instans

Kriteria pencocokan instans, atau kelayakan instans, menentukan instans mana yang diizinkan untuk diluncurkan dan dijalankan di Reservasi Kapasitas. Reservasi Kapasitas dapat memiliki salah satu kriteria pencocokan berikut:

- Buka — Reservasi Kapasitas secara otomatis mencocokkan semua instans yang memiliki atribut yang cocok (jenis instans, platform, dan Zona Ketersediaan). Instans baru dan yang sudah ada yang memiliki atribut yang cocok dijalankan secara otomatis di Reservasi Kapasitas tanpa konfigurasi tambahan apa pun.
- Bertarget — Reservasi Kapasitas hanya menerima instance yang memiliki atribut yang cocok (jenis instans, platform, dan Zona Ketersediaan), dan yang secara eksplisit menargetkan Reservasi Kapasitas. Instans harus secara khusus menargetkan Reservasi Kapasitas untuk diluncurkan atau dijalankan dalam kapasitas yang dicadangkan. Ini memungkinkan Anda untuk secara eksplisit mengontrol instans mana yang diizinkan untuk dijalankan dalam kapasitas yang dicadangkan dan membantu Anda menghindari penggunaan kapasitas cadangan yang tidak disengaja.

Saat meminta Reservasi Kapasitas di masa mendatang, Anda hanya dapat menentukan kriteria pencocokan yang ditargetkan. Ini memastikan bahwa kapasitas yang dikirimkan oleh Reservasi Kapasitas bersifat inkremental, atau tambahan, untuk setiap instans yang sedang berjalan atau kapasitas cadangan yang Anda miliki pada saat pengiriman. Setelah Reservasi Kapasitas aktif di akun Anda, Anda dapat mengubah kriteria pencocokan instans untuk dibuka jika diperlukan.

Namun, perlu diingat bahwa setiap instans yang cocok akan dijalankan secara otomatis di Reservasi Kapasitas, yang dapat menyebabkan penggunaan kapasitas yang tidak disengaja dan mencegah Anda meluncurkan instans baru untuk jumlah instans penuh yang diminta.

Perbedaan antara Reservasi Kapasitas, Instans Terpesan, dan Savings Plans

Tabel berikut menyoroti perbedaan utama antara Reservasi Kapasitas, Instans Terpesan, dan Savings Plans:

| | Reservasi Kapasitas | Instans Terpesan Zonal | Instans Terpesan Regional | Savings Plans |
|---|---|---|---|---|
| Jangka waktu | Tidak ada komitmen yang diperlukan untuk Reservasi Kapasitas penggunaan segera. Mereka dapat dibuat, dimodifikasi, dan dibatalkan sesuai kebutuhan.<br><br>Dengan Reservasi Kapasitas yang akan datang, Anda menentukan durasi komitmen untuk menjaga kapasitas di akun Anda. Anda dapat membatalkan kapan saja; biaya pembatalan mungkin berlaku selama durasi komitmen. | Memerlukan komitmen tetap satu tahun atau tiga tahun | | |

| | Reservasi Kapasitas | Instans Terpesan Zonal | Instans Terpesan Regional | Savings Plans |
|---|---|---|---|---|
| Keuntungan kapasitas | Kapasitas yang terpesan dalam Zona Ketersediaan tertentu. | | Tidak ada kapasitas tersimpan. | |
| Diskon tagihan | Tidak ada diskon penagihan. † | Berikan diskon penagihan. | | |
| Batas Instans | Batas On-Demand Instans Anda per Wilayah berlaku. | Default adalah 20 per Zona Ketersediaan. Anda dapat meminta kenaikan batas. | Default adalah 20 per Wilayah. Anda dapat meminta kenaikan batas. | Tanpa batas. |

† Anda dapat menggabungkan Reservasi Kapasitas dan Savings Plans atau Instans Terpesan regional dengan untuk mendapatkan diskon.

Untuk informasi selengkapnya, lihat berikut ini:

- Ikhtisar Instans Cadangan untuk Amazon EC2
- Panduan Pengguna Savings Plans

Platform yang didukung

Anda harus membuat Reservasi Kapasitas dengan platform yang benar untuk memastikannya cocok dengan instans Anda. Reservasi Kapasitas mendukung nilai-nilai berikut untuk`platform`:

- Linux/UNIX
- Linux dengan SQL Server Standard
- Linux dengan SQL Server Web
- Linux dengan SQL Server Enterprise
- SUSE Linux
- Linux Red Hat Enterprise
- RHEL dengan SQL Server Standard

- RHEL dengan SQL Server Enterprise
- RHEL dengan SQL Server Web
- RHEL dengan HA
- RHEL dengan Standar HA dan SQL Server
- RHEL dengan HA dan SQL Server Enterprise
- Ubuntu Pro
- Windows
- Windows dengan SQL Server
- Windows dengan SQL Server Web
- Windows dengan SQL Server Standard
- Windows dengan SQL Server Enterprise

Untuk memastikan instans berjalan di Reservasi Kapasitas tertentu, platform Reservasi Kapasitas harus sesuai dengan platform AMI yang digunakan untuk meluncurkan instans. Untuk AMI Linux, penting untuk memeriksa apakah platform AMI menggunakan nilai umum Linux/UNIX atau nilai yang lebih spesifik seperti SUSE Linux.

Console

Untuk memeriksa platform AMI

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih AMI.
3. Pilih AMI.
4. Pada tab Detail, catat nilai detail Platform.

AWS CLI

Untuk memeriksa platform AMI

Gunakan perintah describe-images dan periksa nilainya. `PlatformDetails`

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
```

```
    --query Images[*].PlatformDetails
```

Berikut ini adalah output contoh.

```
[
    "Linux/UNIX"
]
```

PowerShell

Untuk memeriksa platform AMI

Gunakan Get-EC2Image cmdlet dan periksa nilainya. `PlatformDetails`

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).PlatformDetails
```

Berikut ini adalah output contoh.

```
Linux/UNIX
```

Kuota

Jumlah instans yang diizinkan untuk memesan kapasitas didasarkan pada kuota On-Demand Instans akun Anda. Anda dapat memesan kapasitas untuk sebanyak mungkin instans sesuai kuota yang diizinkan, dikurangi jumlah instans yang sudah berjalan.

Reservasi Kapasitas di`assessing`,,`scheduled`, `pending`atau `active`, dan `delayed` negara bagian diperhitungkan dalam kuota On-Demand Instans Anda.

Batasan

Sebelum Anda membuat Reservasi Kapasitas, perhatikan batasan dan larangan berikut.

- Reservasi Kapasitas yang aktif dan tidak digunakan dihitung dalam batas On-Demand Instans Anda.
- Reservasi Kapasitas tidak dapat ditransfer dari satu AWS akun ke akun lainnya. Namun, Anda dapat berbagi Reservasi Kapasitas dengan AWS akun lain. Untuk informasi selengkapnya, lihat Reservasi Kapasitas Bersama.
- Diskon penagihan Instans Terpesan zonal tidak berlaku untuk Reservasi Kapasitas.

- Reservasi Kapasitas dapat dibuat dalam grup penempatan klaster. Grup penempatan partisi dan tersebar tidak didukung.
- Reservasi Kapasitas tidak dapat digunakan dengan Host Khusus. Reservasi Kapasitas dapat digunakan dengan Instans Khusus.
- [Instans Windows] Reservasi Kapasitas tidak dapat digunakan dengan Bring Your Own License (BYOL).
- [Instans Red Hat] Reservasi Kapasitas dapat digunakan dengan Bring Your Own License (BYOL).
- Reservasi Kapasitas tidak memastikan bahwa instans yang dihibernasi dapat melanjutkan setelah Anda mencoba untuk memulainya.
- Anda dapat meminta Reservasi Kapasitas tanggal mendatang untuk jumlah instans dengan minimal 32 vCPU. Misalnya, jika Anda meminta Reservasi Kapasitas yang akan datang untuk `m5.xlarge` instans, Anda harus meminta setidaknya 8 instans (8 * m5.xlarge = 32 vCPU).
- Anda dapat meminta Reservasi Kapasitas tanggal mendatang untuk jenis instance dalam keluarga berikut: C, G, I, M, R, T, U, dan X.

## Harga dan penagihan Reservasi Kapasitas

Topik di bagian ini memberikan gambaran umum tentang harga dan penagihan untuk Reservasi Kapasitas.

Topik

- Harga
- Penagihan
- Diskon tagihan
- Biaya pembatalan di Laporan Biaya dan Penggunaan 2.0
- Melihat tagihan Anda

### Harga

Reservasi Kapasitas dikenakan tarif yang setara On-Demand baik Anda menjalankan instans dalam kapasitas cadangan atau tidak, termasuk biaya tambahan regional yang berlaku untuk Instans Khusus. Jika Anda tidak menggunakan reservasi, ini akan muncul sebagai reservasi yang tidak terpakai di tagihan Amazon EC2 Anda. Saat Anda menjalankan instans yang cocok dengan atribut reservasi, Anda cukup membayar untuk instans tersebut dan tidak membayar apa pun untuk reservasi. Tidak ada biaya di muka atau biaya tambahan.

Misalnya, jika Anda membuat Reservasi Kapasitas untuk 20 instans Linux `m4.large` dan menjalankan 15 instans Linux `m4.large` di Zona Ketersediaan yang sama, Anda akan dikenai biaya sebesar 15 instans aktif dan 5 instans yang tidak digunakan dalam reservasi.

Diskon tagihan untuk Savings Plans dan Instans Terpesan Regional berlaku untuk Reservasi Kapasitas. Untuk informasi selengkapnya, lihat Diskon tagihan.

Untuk informasi selengkapnya, lihat Penetapan Harga Amazon EC2.

Penagihan

Tagihan dimulai segera setelah Reservasi Kapasitas disediakan di akun Anda, dan tagihan berlanjut selama Reservasi Kapasitas tetap disediakan di akun Anda. Untuk Reservasi Kapasitas yang akan datang, ini berarti penagihan dimulai hanya setelah Reservasi Kapasitas disediakan di akun Anda pada tanggal mendatang yang diminta.

Reservasi Kapasitas ditagih dengan perincian per detik, dengan minimal 60 detik. Ini berarti bahwa Anda akan dikenai biaya untuk sebagian jam. Misalnya, jika Reservasi Kapasitas tetap disediakan di akun Anda selama 24 jam dan 15 menit, Anda ditagih untuk `24.25` jam reservasi.

Contoh berikut menunjukkan bagaimana Reservasi Kapasitas ditagih. Reservasi Kapasitas dibuat untuk satu instance `m4.large` Linux, yang memiliki On-Demand tarif $0,10 per jam penggunaan. Dalam contoh ini, Reservasi Kapasitas disediakan di akun selama lima jam. Reservasi Kapasitas tidak digunakan untuk jam pertama, sehingga ditagih untuk satu jam yang tidak digunakan dengan On-Demand tarif standar jenis `m4.large` instans. Dalam jam dua sampai lima, Reservasi Kapasitas ditempati oleh instans `m4.large`. Selama itu, Reservasi Kapasitas tidak mengakumulasi biaya, tetapi akun ditagih untuk instans `m4.large` yang menempatinya. Pada jam keenam, Reservasi Kapasitas dibatalkan dan instans `m4.large` berjalan normal di luar kapasitas terpesan. Untuk jam itu, dikenakan biaya pada On-Demand tingkat jenis `m4.large` instance.

| Hour | 1 | 2 | 3 | 4 | 5 | 6 | Total cost |
|---|---|---|---|---|---|---|---|
| **Unused Capacity Reservation** | $0.10 | $0.00 | $0.00 | $0.00 | $0.00 | $0.00 | **$0.10** |
| **On-demand Instance Usage** | $0.00 | $0.10 | $0.10 | $0.10 | $0.10 | $0.10 | **$0.50** |
| **Hourly cost** | **$0.10** | **$0.10** | **$0.10** | **$0.10** | **$0.10** | **$0.10** | **$0.60** |

Diskon tagihan

Diskon penagihan untuk  Paket Tabungan dan Inst  ans Cadangan Regional berlaku untuk Reservasi Kapasitas. AWS secara otomatis menerapkan diskon ini ke Reservasi Kapasitas yang memiliki

atribut yang cocok. Saat Reservasi Kapasitas digunakan oleh sebuah instans, diskon diterapkan ke instans tersebut. Diskon secara istimewa diterapkan pada penggunaan instans sebelum mencakup Reservasi Kapasitas yang tidak digunakan.

Diskon tagihan Instans Terpesan zonal tidak berlaku untuk Reservasi Kapasitas.

Untuk informasi selengkapnya, lihat berikut ini:

- Ikhtisar Instans Cadangan untuk Amazon EC2
- Panduan Pengguna Savings Plans
- Opsi tagihan dan pembelian

Biaya pembatalan di Laporan Biaya dan Penggunaan 2.0

Jika Anda membatalkan Reservasi Kapasitas di masa mendatang dan dikenakan biaya pembatalan, biaya tersebut akan muncul di Laporan AWS Biaya dan Penggunaan (CUR) 2.0 Anda. Untuk informasi selengkapnya tentang kolom reservasi kapasitas di CUR 2.0, lihat Kolom reservasi kapasitas. Tabel berikut menunjukkan bagaimana item baris Reservasi Kapasitas muncul sebelum dan sesudah pembatalan.

Nilai CUR 2.0 untuk reservasi aktif

| kapasitas_reservasi_kapasitas_reservasi_status | tipe_penggunaan_baris_item_baris_item | Apa yang diwakilinya |
|---|---|---|
| `Reserved` | Reservasi | Total kapasitas yang dicadangkan |
| `Used` | BoxUsage | Instans yang diluncurkan |
| `Unused` | UnusedBox | Kapasitas idle |

Nilai CUR 2.0 setelah pembatalan

| kapasitas_reservasi_kapasitas_reservasi_status | tipe_penggunaan_baris_item_baris_item | Apa yang diwakilinya |
|---|---|---|
| `Cancelling` | Reservasi | Kapasitas yang dibatalkan |
| `Cancelling` | UnusedBox | Biaya pembatalan |

### Melihat tagihan Anda

Anda dapat meninjau tagihan dan biaya untuk akun Anda di konsol AWS Manajemen Penagihan dan Biaya .

- Dasbor menampilkan ringkasan pengeluaran untuk akun Anda.
- Pada halaman Tagihan, pada Detail, perluas bagian Elastic Compute Cloud dan Wilayah untuk mendapatkan informasi tagihan terkait Kapasitas Terpesan Anda.

Anda dapat melihat tagihannya secara online, atau Anda dapat mengunduh file CSV. Untuk informasi selengkapnya, lihat I tem baris Reservasi Kapasitas.

### Membuat Reservasi Kapasitas

Anda dapat membuat Reservasi Kapasitas kapan saja untuk memastikan bahwa Anda memiliki kapasitas komputasi yang tersedia di Zona Ketersediaan tertentu. Reservasi Kapasitas dapat dimulai segera, atau dapat dimulai di masa mendatang. Kapasitas menjadi tersedia untuk digunakan hanya setelah Reservasi Kapasitas memasuki `active` negara bagian.

> **Note**
>
> Jika Anda membuat Reservasi Kapasitas dengan kriteria `open` pencocokan instans, dan Anda menjalankan instans dengan atribut yang cocok pada saat Reservasi Kapasitas aktif, instans tersebut secara otomatis berjalan dalam kapasitas yang dicadangkan. Untuk menghindari hal ini, gunakan kriteria `targeted` pencocokan instance. Untuk informasi selengkapnya, lihat Kriteria pencocokan instans.

Permintaan Anda untuk membuat Reservasi Kapasitas bisa gagal jika salah satu dari yang berikut ini benar:

- Amazon EC2 tidak memiliki kapasitas yang cukup untuk memenuhi permintaan. Coba lagi nanti, coba Zona Ketersediaan yang berbeda, atau coba permintaan yang lebih kecil. Jika aplikasi Anda fleksibel di semua tipe dan ukuran instans, coba atribut instans yang berbeda.
- Kuantitas yang diminta melebihi batas On-Demand Instans untuk keluarga instans yang dipilih. Tingkatkan batas On-Demand Instans untuk keluarga instans dan coba lagi. Untuk informasi selengkapnya, lihat Kuota Instans Sesuai Permintaan.

Topik


- Buat Reservasi Kapasitas untuk segera digunakan
- Membuat Reservasi Kapasitas yang akan datang


## Buat Reservasi Kapasitas untuk segera digunakan

Anda membuat Reservasi Kapasitas untuk digunakan segera.

Console

Untuk membuat Reservasi Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Reservasi Kapasitas, lalu pilih Buat Reservasi Kapasitas.
3. Konfigurasikan pengaturan berikut di bagian rincian Instance.
   a. Jenis Instance — Jenis instans yang harus dicadangkan kapasitas.
   b. Platform — Sistem operasi untuk instans Anda. Untuk informasi selengkapnya, lihat Platform yang didukung.
   c. Zona Ketersediaan — Zona Ketersediaan tempat untuk memesan kapasitas.
   d. Sewa — Jenis sewa yang digunakan untuk kapasitas yang dicadangkan. Pilih Default untuk mencadangkan kapasitas pada perangkat keras bersama, atau Didedikasikan untuk cadangan kapasitas pada perangkat keras yang didedikasikan untuk akun Anda.
   e. (Opsional) Grup penempatan ARN — ARN dari grup penempatan cluster tempat membuat Reservasi Kapasitas. Untuk informasi selengkapnya, lihat Gunakan Reservasi Kapasitas dengan grup penempatan.
   f. Jumlah instans total — Jumlah instans yang harus dipesan kapasitas. Jika Anda menentukan jumlah yang melebihi kuota On-Demand Instans yang tersisa untuk jenis instans yang dipilih, permintaan gagal.
4. Konfigurasikan pengaturan berikut di bagian Detail reservasi:
   a. Reservasi Kapasitas dimulai — Pilih Segera.
   b. Reservasi Kapasitas berakhir — Pilih salah satu opsi berikut:
      - Secara manual — Cadangkan kapasitas sampai Anda secara eksplisit membatalkannya.

- Waktu tertentu — Batalkan reservasi kapasitas secara otomatis pada tanggal dan waktu yang ditentukan.

c. Kelayakan instans — Pilih salah satu opsi berikut:

- buka — (Default) Reservasi Kapasitas cocok dengan instans apa pun yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa). Jika Anda meluncurkan sebuah instans dengan atribut yang cocok, atribut ditempatkan ke dalam kapasitas terpesan secara otomatis.
- bertarget — Reservasi Kapasitas hanya menerima instans yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa), dan yang secara eksplisit menargetkan reservasi.

5. Pilih Buat.

AWS CLI

Untuk membuat Reservasi Kapasitas

Gunakan perintah create-capacity-reservation.

```
aws ec2 create-capacity-reservation \
    --availability-zone az_name \
    --instance-type instance_type \
    --instance-count number_of_instances \
    --instance-platform operating_system \
    --instance-match-criteria open|targeted
```

PowerShell

Untuk membuat Reservasi Kapasitas

Gunakan Add-EC2CapacityReservation cmdlet.

```
Add-EC2CapacityReservation `
    -AvailabilityZone az_name `
    -InstanceType instance_type `
    -InstanceCount number_of_instances `
    -InstancePlatform operating_system `
    -InstanceMatchCriterion open|targeted
```

## Membuat Reservasi Kapasitas yang akan datang

Minta Reservasi Kapasitas di masa mendatang jika Anda memerlukan kapasitas yang dipesan agar tersedia pada tanggal dan waktu mendatang.

Setelah Anda meminta Reservasi Kapasitas di masa mendatang, permintaan akan menjalani penilaian untuk menentukan apakah reservasi tersebut dapat didukung. Untuk informasi selengkapnya, lihat Future-dated Penilaian Reservasi Kapasitas.

### Pertimbangan

- Anda dapat meminta Reservasi Kapasitas tanggal mendatang untuk jenis instance dalam keluarga berikut: C, G, I, M, R, T, U, dan X.
- Anda dapat meminta Reservasi Kapasitas tanggal mendatang untuk jumlah instans dengan minimal 32 vCPU. Misalnya, jika Anda meminta Reservasi Kapasitas yang akan datang untuk `m5.xlarge` instans, Anda harus meminta kapasitas untuk setidaknya 8 instans (8 * m5.xlarge = 32 vCPU).
- Anda dapat meminta Reservasi Kapasitas di masa mendatang antara 5 dan 120 hari sebelumnya. Kami menyarankan Anda memintanya setidaknya 56 hari (8 minggu) sebelumnya untuk meningkatkan dukungan.
- Durasi komitmen minimum adalah 14 hari untuk contoh C, G, I, M, R, dan T. Untuk contoh U dan X, durasi komitmen minimum adalah 12 minggu.

### Console

Untuk membuat Reservasi Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Reservasi Kapasitas, lalu pilih Buat Reservasi Kapasitas.
3. Konfigurasikan pengaturan berikut di bagian Detail Instance.
   a. Jenis Inst ance — Jenis instans untuk cadangan kapasitas.
   b. Platform — Sistem operasi untuk instans Anda. Untuk informasi selengkapnya, lihat Platform yang didukung.
   c. Zona Ketersediaan — Zona Ketersediaan tempat untuk memesan kapasitas.

d. Sewa — Jenis sewa yang digunakan untuk kapasitas yang dicadangkan. Pilih Default untuk mencadangkan kapasitas pada perangkat keras bersama, atau Didedikasikan untuk cadangan kapasitas pada perangkat keras yang didedikasikan untuk akun Anda.

e. Jumlah instans total — Jumlah instans yang harus dipesan kapasitas. Jika Anda menentukan jumlah yang melebihi kuota On-Demand Instans yang tersisa untuk jenis instans yang dipilih, permintaan gagal.

4. Konfigurasikan pengaturan berikut di bagian Detail reservasi:

a. Reservasi Kapasitas dimulai — Pilih Pada waktu tertentu.

b. Tanggal Mulai — Tentukan tanggal dan waktu di mana Reservasi Kapasitas harus tersedia untuk digunakan. Untuk informasi selengkapnya, lihat Tanggal dan waktu mulai.

c. Durasi komitmen — Tentukan durasi minimum untuk mana Anda berkomitmen untuk menjaga Reservasi Kapasitas setelah dikirimkan. Untuk informasi selengkapnya, lihat Durasi komitmen.

d. Reservasi Kapasitas berakhir — Pilih salah satu opsi berikut:

- Ketika saya membatalkannya - Cadangkan kapasitas sampai Anda secara eksplisit membatalkannya.
- Waktu tertentu — Batalkan reservasi kapasitas secara otomatis pada tanggal dan waktu yang ditentukan.

5. Pilih Buat.

AWS CLI

Untuk membuat Reservasi Kapasitas

Gunakan perintah create-capacity-reservation.

```
aws ec2 create-capacity-reservation \
    --availability-zone az_name \
    --instance-type instance_type \
    --instance-count number_of_instances \
    --instance-platform operating_system \
    --instance-match-criteria targeted \
    --delivery-preference incremental \
    --commitment-duration commitment_in_seconds \
    --start-date YYYY-MMDDThh:mm:ss.sssZ
```

PowerShell

Untuk membuat Reservasi Kapasitas

Gunakan Add-EC2CapacityReservation cmdlet.

```
Add-EC2CapacityReservation `
    -AvailabilityZone az_name `
    -InstanceType instance_type `
    -InstanceCount number_of_instances `
    -InstancePlatform operating_system `
    -InstanceMatchCriterion targeted `
    -DeliveryPreference incremental `
    -CommitmentDuration commitment_in_seconds `
    -StartDate  YYYY-MMDDThh:mm:ss.sssZ
```

Melihat status Reservasi Kapasitas

Amazon EC2 terus memantau status Reservasi Kapasitas Anda.

Karena model konsist ensi akhirnya diikuti oleh Amazon EC2 API, setelah Anda membuat Reservasi Kapasitas, diperlukan waktu hingga 5 menit agar status Reservasi Kapasitas mencerminkan keadaan tersebut`active`. Selama waktu ini, Reservasi Kapasitas mungkin tetap di `pending` negara bagian. Namun, itu mungkin sudah tersedia untuk digunakan, dalam hal ini upaya untuk meluncurkan instance ke Reservasi Kapasitas akan berhasil.

Console

Untuk melihat Reservasi Kapasitas Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Reservasi Kapasitas.

AWS CLI

Untuk menjelaskan Reservasi Kapasitas Anda

Gunakan perintah describe-capacity-reservations:

Misalnya, perintah berikut menjelaskan semua Reservasi Kapasitas.

```
aws ec2 describe-capacity-reservations
```

Berikut ini adalah output contoh.

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-1234abcd56EXAMPLE",
            "EndDateType": "unlimited",
            "AvailabilityZone": "eu-west-1a",
            "InstanceMatchCriteria": "open",
            "Tags": [],
            "EphemeralStorage": false,
            "CreateDate": "2019-08-16T09:03:18.000Z",
            "AvailableInstanceCount": 1,
            "InstancePlatform": "Linux/UNIX",
            "TotalInstanceCount": 1,
            "State": "active",
            "Tenancy": "default",
            "EbsOptimized": true,
            "InstanceType": "a1.medium",
            "PlacementGroupArn": "arn:aws:ec2:us-east-1:123456789012:placement-
group/MyPG"
        },
        {
            "CapacityReservationId": "cr-abcdEXAMPLE9876ef",
            "EndDateType": "unlimited",
            "AvailabilityZone": "eu-west-1a",
            "InstanceMatchCriteria": "open",
            "Tags": [],
            "EphemeralStorage": false,
            "CreateDate": "2019-08-07T11:34:19.000Z",
            "AvailableInstanceCount": 3,
            "InstancePlatform": "Linux/UNIX",
            "TotalInstanceCount": 3,
            "State": "cancelled",
            "Tenancy": "default",
            "EbsOptimized": true,
            "InstanceType": "m5.large"
        }
    ]
```

```
}
```

PowerShell

Untuk mendeskripsikan Reservasi Kapasitas

Gunakan Get-EC2CapacityReservation cmdlet.

```
Get-EC2CapacityReservation `
    -CapacityReservationId cr-1234abcd56EXAMPLE
```

### Status Reservasi Kapasitas

Reservasi Kapasitas memiliki kemungkinan status berikut.

| Status | Deskripsi |
| --- | --- |
| `active` | Kapasitas tersedia untuk digunakan. |
| `expired` | Reservasi Kapasitas berakhir secara otomatis pada tanggal dan waktu yang ditentukan dalam permintaan reservasi Anda. Kapasitas terpesan tidak lagi tersedia untuk Anda gunakan. |
| `cancelled` | Reservasi Kapasitas dibatalkan. Kapasitas terpesan tidak lagi tersedia untuk Anda gunakan. |
| `pending` | Permintaan Reservasi Kapasitas berhasil tetapi penyediaan kapasitas masih tertunda. |
| `failed` | Permintaan Reservasi Kapasitas gagal. Permintaan dapat gagal karena parameter permintaan yang tidak valid, batasan kapasitas, atau batasan batas instans. Anda dapat melihat permintaan yang gagal selama 60 menit. |
| `scheduled` | (Hanya Future-dated Reservasi Kapasitas) Permintaan Reservasi Kapasitas di masa mendatang telah disetujui dan Reservasi Kapasitas dijadwalkan untuk pengiriman pada tanggal mulai yang diminta. |
| `assessing` | (Hanya Future-dated Reservasi Kapasitas) Amazon EC2 sedang menilai permintaan Anda untuk Reservasi Kapasitas di masa mendatang. |

| Status | Deskripsi |
| --- | --- |
| | Untuk informasi selengkapnya, lihat Future-dated Penilaian Reservasi Kapasitas. |
| `delayed` | (Hanya Future-dated Reservasi Kapasitas) Amazon EC2 mengalami keterlambatan dalam penyediaan Reservasi Kapasitas tanggal mendatang yang diminta. Amazon EC2 tidak dapat mengirimkan kapasitas yang diminta pada tanggal dan waktu mulai yang diminta. |
| `unsupported` | (Future-dated Hanya Reservasi Kapasitas) Amazon EC2 tidak dapat mendukung permintaan Reservasi Kapasitas yang akan datang karena keterbatasan kapasitas. Anda dapat melihat permintaan yang tidak didukung selama 30 hari. Reservasi Kapasitas tidak akan dikirimkan. |
| `cancelling` | (Future-dated Hanya Reservasi Kapasitas) Reservasi Kapasitas sedang dibatalkan. Kapasitas telah dilepaskan tetapi biaya berlanjut hingga akhir periode komitmen yang dikurangi. Transisi Reservasi Kapasitas ke `cancelled` saat komitmen berakhir. Untuk informasi selengkapnya, lihat Biaya pembatalan. |

Luncurkan instans ke dalam Reservasi Kapasitas yang ada

Anda hanya dapat meluncurkan instance ke Reservasi Kapasitas yang:

- Memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa)
- Memiliki kapasitas yang cukup
- Adalah di `active` negara bagian

Saat Anda meluncurkan sebuah instans, Anda dapat menentukan apakah akan meluncurkan instans tersebut ke salah satu Reservasi Kapasitas `open`, ke dalam Reservasi Kapasitas tertentu, atau ke dalam kelompok Reservasi Kapasitas.

Atau, Anda dapat mengonfigurasi instans agar tidak berjalan di Reservasi Kapasitas, meskipun Anda memiliki Reservasi Kapasitas `open` yang cocok dengan atribut dan kapasitas yang tersedia.

Meluncurkan sebuah instans ke dalam Reservasi Kapasitas mengurangi kapasitasnya yang tersedia dengan jumlah instans yang diluncurkan. Misalnya, jika Anda meluncurkan tiga instans, kapasitas Reservasi Kapasitas yang tersedia dikurangi tiga.

Console

Untuk meluncurkan instans ke Reservasi Kapasitas yang ada

1. Ikuti prosedur untuk meluncurkan instance, tetapi jangan meluncurkan instans sampai Anda menyelesaikan langkah-langkah berikut untuk menentukan pengaturan untuk grup penempatan dan Reservasi Kapasitas.
2. Per luas detail lanj utan dan lakukan hal berikut:

   a. Untuk grup Pen empatan, pilih grup penempatan cluster tempat peluncuran instans.

   b. Untuk Reservasi Kapasitas, pilih salah satu opsi berikut, tergantung pada konfigurasi Reservasi Kapasitas:

      - Tidak ada — Mencegah instans diluncurkan ke Reservasi Kapasitas. Instans berjalan dalam On-Demand kapasitas.
      - Buka — Meluncurkan instance ke Reservasi Kapasitas apa pun yang memiliki atribut yang cocok dan kapasitas yang cukup untuk jumlah instans yang Anda pilih. Jika tidak ada Reservasi Kapasitas yang cocok dengan kapasitas yang memadai, instans menggunakan On-Demand kapasitas.
      - Tentukan Reservasi Kapasitas — Meluncurkan instance ke Reservasi Kapasitas yang dipilih. Jika Reservasi Kapasitas yang dipilih tidak memiliki kapasitas yang cukup untuk jumlah instans yang Anda pilih, peluncuran instans akan gagal.
      - Tentukan Grup Sumber Daya Reservasi Kapasitas — Meluncurkan instance ke Reservasi Kapasitas apa pun dengan atribut yang cocok dan kapasitas yang tersedia di grup Reservasi Kapasitas yang dipilih. Jika grup yang dipilih tidak memiliki Reservasi Kapasitas dengan atribut yang cocok dan kapasitas yang tersedia, instans akan diluncurkan ke On-Demand kapasitas.
      - Tentukan Reservasi Kapasitas saja — Meluncurkan instance ke Reservasi Kapasitas. Jika ID Reservasi Kapasitas tidak ditentukan, instans akan diluncurkan ke Reservasi Kapasitas terbuka. Jika kapasitas tidak tersedia, instans gagal diluncurkan.
      - Tentukan hanya Grup Sumber Daya Reservasi Kapasitas — Meluncurkan instance ke Reservasi Kapasitas di Grup Sumber Daya Reservasi Kapasitas. Jika ARN Grup

Sumber Daya Reservasi Kapasitas tidak ditentukan, instans akan diluncurkan ke Reservasi Kapasitas terbuka. Jika kapasitas tidak tersedia, instans gagal diluncurkan.

3. Di panel Ringkasan, tinjau konfigurasi instans Anda, lalu pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan instans ke Reservasi Kapasitas yang ada

Gunakan perintah run-instance dan tentukan opsi. `--capacity-reservation-specification`

Contoh berikut meluncurkan instance ke Reservasi Kapasitas terbuka dengan atribut yang cocok dan kapasitas yang tersedia:

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count 1 \
    --instance-type t2.micro \
    --key-name my-key-pair \
    --subnet-id subnet-0abcdef1234567890 \
    --capacity-reservation-specification CapacityReservationPreference=open
```

Contoh berikut meluncurkan instance ke Reservasi `targeted` Kapasitas:

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count 1 \
    --instance-type t2.micro \
    --key-name my-key-pair \
    --subnet-id subnet-0abcdef1234567890 \
    --capacity-reservation-specification \
        CapacityReservationTarget={CapacityReservationId=cr-1234abcd56EXAMPLE}
```

Contoh berikut meluncurkan instance ke grup Reservasi Kapasitas yang ditentukan:

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count 1 \
    --instance-type t2.micro \
    --key-name my-key-pair \
    --subnet-id subnet-0abcdef1234567890 \
```

```
    --capacity-reservation-specification \

 CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-
groups:us-west-2:123456789012:group/my-cr-group}
```

Contoh berikut meluncurkan instance ke Reservasi Kapasitas saja. Karena tidak menentukan ID Reservasi Kapasitas, instans diluncurkan di Reservasi Kapasitas terbuka apa pun dengan atribut yang cocok dan kapasitas yang tersedia:

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count 1 \
    --instance-type t2.micro \
    --key-name my-key-pair \
    --subnet-id subnet-0abcdef1234567890 \
    --capacity-reservation-specification \
        CapacityReservationPreference=capacity-reservations-only
```

Contoh berikut meluncurkan instance ke Reservasi Kapasitas tertentu saja. Jika kapasitas tidak tersedia di Reservasi Kapasitas yang ditentukan, instans gagal diluncurkan.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count 1 \
    --instance-type t2.micro \
    --key-name my-key-pair \
    --subnet-id subnet-0abcdef1234567890 \
    --capacity-reservation-specification \
        CapacityReservationPreference=capacity-reservations-only \
        CapacityReservationTarget={CapacityReservationId=cr-1234abcd56EXAMPLE}
```

PowerShell

Untuk meluncurkan instans ke Reservasi Kapasitas yang ada

Gunakan New-EC2Instance cmdlet.

Contoh berikut meluncurkan instance ke Reservasi Kapasitas terbuka dengan atribut yang cocok dan kapasitas yang tersedia:

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
```

```
    -InstanceType t2.micro `
    -KeyName "my-key-pair" `
    -SubnetId subnet-0abcdef1234567890 `
    -CapacityReservationSpecification_CapacityReservationPreference "open"
```

Contoh berikut meluncurkan instance ke Reservasi `targeted` Kapasitas:

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType t2.micro `
    -KeyName "my-key-pair" `
    -SubnetId subnet-0abcdef1234567890 `
    -CapacityReservationTarget_CapacityReservationId cr-1234abcd56EXAMPLE
```

Contoh berikut meluncurkan instance ke grup Reservasi Kapasitas yang ditentukan:

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType t2.micro `
    -KeyName "my-key-pair" `
    -SubnetId subnet-0abcdef1234567890 `
    -CapacityReservationTarget_CapacityReservationResourceGroupArn `
        "arn:aws:resource-groups:us-west-2:123456789012:group/my-cr-group"
```

Contoh berikut meluncurkan instance ke Reservasi Kapasitas saja. Karena tidak menentukan ID Reservasi Kapasitas, instans diluncurkan di Reservasi Kapasitas terbuka apa pun dengan atribut yang cocok dan kapasitas yang tersedia:

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType t2.micro `
    -KeyName "my-key-pair" `
    -SubnetId subnet-0abcdef1234567890 `
    -CapacityReservationSpecification_CapacityReservationPreference "capacity-
reservations-only"
```

Contoh berikut meluncurkan instance ke Reservasi Kapasitas tertentu saja. Jika kapasitas tidak tersedia di Reservasi Kapasitas yang ditentukan, instans gagal diluncurkan.

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
```

```
    -InstanceType t2.micro `
    -KeyName "my-key-pair" `
    -SubnetId subnet-0abcdef1234567890 `
    -CapacityReservationSpecification_CapacityReservationPreference "capacity-
reservations-only" `
    -CapacityReservationTarget_CapacityReservationId cr-1234abcd56EXAMPLE
```

### Mengubah Reservasi Kapasitas yang aktif

Jika Anda memiliki Reservasi Kapasitas yang sudah ada yang tidak sesuai dengan beban kerja yang membutuhkan kapasitas, Anda dapat mengubah jumlah instans, kelayakan instans (`open`atau`targeted`), dan waktu akhir (`At specific time`atau`Manually`). Jika Anda menentukan jumlah instans baru yang melebihi batas On-Demand Instans yang tersisa untuk jenis instans yang dipilih, pembaruan gagal.

Modifikasi yang diizinkan tergantung pada status Reservasi Kapasitas:

- `assessing`atau `scheduled` status — Anda dapat memodifikasi tag saja.
- `pending`status — Anda tidak dapat mengubah Reservasi Kapasitas dengan cara apa pun.
- `active`status tetapi masih dalam durasi komitmen — Anda tidak dapat mengurangi jumlah instans di bawah jumlah instans yang dikomit, atau menetapkan tanggal akhir sebelum durasi komit. Semua modifikasi lainnya diperbolehkan.
- `active`status tanpa durasi komitmen atau durasi komitmen yang telah berlalu — Semua modifikasi diperbolehkan.
- `expired`,`cancelled`,`unsupported`, atau `failed` negara — Anda tidak dapat mengubah Reservasi Kapasitas dengan cara apa pun.

### Pertimbangan-pertimbangan

- Anda tidak dapat mengubah jenis instans, platform, Zona Ketersediaan, atau sewa setelah pembuatan. Jika Anda perlu mengubah salah satu atribut ini, kami menyarankan Anda untuk membatalkan reservasi, dan kemudian membuat yang baru dengan atribut yang diperlukan.
- Jika Anda mengubah Reservasi Kapasitas yang ada dengan mengubah kelayakan instans dari `targeted` keopen, setiap instans yang berjalan yang cocok dengan atribut Reservasi Kapasitas, memiliki `CapacityReservationPreference` parameter yang disetel keopen, dan belum berjalan dalam Reservasi Kapasitas, akan secara otomatis menggunakan Reservasi Kapasitas yang dimodifikasi.

- Untuk mengubah kelayakan instans, Reservasi Kapasitas harus benar-benar tidak digunakan (penggunaan nol) karena Amazon EC2 tidak dapat mengubah kelayakan instans saat instans berjalan di dalam reservasi.

Console

Untuk mengubah Reservasi Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Reservasi Kapasitas, pilih Reservasi Kapasitas yang akan dimodifikasi, lalu pilih Edit.
3. Ubah opsi Kapasitas Total, Akhir Reservasi Kapasitas, atau opsi kelayakan Instans sesuai kebutuhan, lalu pilih Simpan.

AWS CLI

Untuk mengubah Reservasi Kapasitas

Gunakan perintah modify- capacity-reservation. Contoh berikut memodifikasi Reservasi Kapasitas yang ditentukan untuk mencadangkan kapasitas untuk delapan instans.

```
aws ec2 modify-capacity-reservation \
    --capacity-reservation-id cr-1234567890abcdef0 \
    --instance-count 8
```

PowerShell

Untuk mengubah Reservasi Kapasitas

Gunakan Edit-EC2CapacityReservation cmdlet. Contoh berikut memodifikasi Reservasi Kapasitas yang ditentukan untuk mencadangkan kapasitas untuk delapan instans.

```
Edit-EC2CapacityReservation `
    -CapacityReservationId cr-1234567890abcdef0 `
    -InstanceCount 8
```

Mengubah setelan Reservasi Kapasitas instans Anda

Anda dapat memodifikasi pengaturan Reservasi Kapasitas berikut untuk instans yang berhenti kapan saja:

- Mulai di Reservasi Kapasitas apa pun yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa) dan kapasitas yang tersedia.
- Mulai instans di Reservasi Kapasitas tertentu.
- Mulailah di Reservasi Kapasitas apa pun yang memiliki kecocokan atribut dan ketersediaan kapasitas di grup Reservasi Kapasitas
- Mencegah instans dimulai dalam Reservasi Kapasitas.

Console

Untuk mengubah setelan Reservasi Kapasitas instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Instans dan pilih instans yang akan dimodifikasi. Hentikan instans jika belum dihentikan.
3. Pilih T indakan, Peng aturan Instans, Ubah Pengaturan Reservasi Kapasitas.
4. Untuk Reservasi Kapasitas, pilih salah satu opsi berikut:
   - Buka — Meluncurkan instance ke Reservasi Kapasitas apa pun yang memiliki atribut yang cocok dan kapasitas yang cukup untuk jumlah instans yang Anda pilih. Jika tidak ada Reservasi Kapasitas yang cocok dengan kapasitas yang memadai, instans menggunakan On-Demand kapasitas.
   - Tidak ada — Mencegah instans diluncurkan ke Reservasi Kapasitas. Instans berjalan dalam On-Demand kapasitas.
   - Tentukan Reservasi Kapasitas — Meluncurkan instance ke Reservasi Kapasitas yang dipilih. Jika Reservasi Kapasitas yang dipilih tidak memiliki kapasitas yang cukup untuk jumlah instans yang Anda pilih, peluncuran instans akan gagal.
   - Tentukan grup Reservasi Kapasitas — Meluncurkan instance ke Reservasi Kapasitas apa pun dengan atribut yang cocok dan kapasitas yang tersedia di grup Reservasi Kapasitas yang dipilih. Jika grup yang dipilih tidak memiliki Reservasi Kapasitas dengan atribut yang cocok dan kapasitas yang tersedia, instans akan diluncurkan ke On-Demand kapasitas.
   - Tentukan Reservasi Kapasitas saja — Meluncurkan instance ke Reservasi Kapasitas. Jika ID Reservasi Kapasitas tidak ditentukan, instans akan diluncurkan ke Reservasi Kapasitas terbuka. Jika kapasitas tidak tersedia, instans gagal diluncurkan.
   - Tentukan hanya Grup Sumber Daya Reservasi Kapasitas — Meluncurkan instance ke Reservasi Kapasitas di Grup Sumber Daya Reservasi Kapasitas. Jika ARN Grup Sumber

Daya Reservasi Kapasitas tidak ditentukan, instans akan diluncurkan ke Reservasi Kapasitas terbuka. Jika kapasitas tidak tersedia, instans gagal diluncurkan.

AWS CLI

Untuk mengubah setelan Reservasi Kapasitas instans

Gunakan perintah modify-instance-capacity-reservation-attributes.

Contoh berikut mengubah preferensi Reservasi Kapasitas menjadinone.

```
aws ec2 modify-instance-capacity-reservation-attributes \
    --instance-id i-1234567890abcdef0 \
    --capacity-reservation-specification CapacityReservationPreference=none
```

Berikut contoh target untuk Reservasi Kapasitas tertentu.

```
aws ec2 modify-instance-capacity-reservation-attributes \
    --instance-id i-1234567890abcdef0 \
    --capacity-reservation-specification \
    CapacityReservationTarget={CapacityReservationId=cr-1234567890abcdef0}
```

Contoh berikut mengubah target ke grup Reservasi Kapasitas tertentu.

```
aws ec2 modify-instance-capacity-reservation-attributes \
    --instance-id i-1234567890abcdef0 \
    --capacity-reservation-specification \

 CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-
groups:us-west-2:123456789012:group/my-cr-group}
```

Contoh berikut mengubah preferensi Reservasi Kapasitas menjadicapacity-reservation-only. Karena tidak menentukan Reservasi Kapasitas, instans diluncurkan ke Reservasi Kapasitas terbuka apa pun dengan atribut yang cocok dan kapasitas yang tersedia.

```
aws ec2 modify-instance-capacity-reservation-attributes \
    --instance-id i-1234567890abcdef0 \
    --capacity-reservation-specification CapacityReservationPreference=capacity-
reservation-only
```

Contoh berikut mengubah preferensi Reservasi Kapasitas ke `capacity-reservation-only` dan mengubah target ke Reservasi Kapasitas tertentu. Jika kapasitas tidak tersedia di Reservasi Kapasitas yang ditentukan, instans gagal diluncurkan.

```
aws ec2 modify-instance-capacity-reservation-attributes \
    --instance-id i-1234567890abcdef0 \
    --capacity-reservation-specification \
        CapacityReservationPreference=capacity-reservation-only \
        CapacityReservationTarget={CapacityReservationId=cr-1234567890abcdef0}
```

PowerShell

Untuk mengubah setelan Reservasi Kapasitas instans

Gunakan Edit-EC2InstanceCapacityReservationAttribute cmdlet.

Contoh berikut mengubah preferensi Reservasi Kapasitas menjadinone.

```
Edit-EC2InstanceCapacityReservationAttribute `
    -InstanceId i-1234567890abcdef0 `
    -CapacityReservationSpecification_CapacityReservationPreference "none"
```

Berikut contoh target untuk Reservasi Kapasitas tertentu.

```
Edit-EC2InstanceCapacityReservationAttribute `
    -InstanceId i-1234567890abcdef0 `
    -CapacityReservationTarget_CapacityReservationId cr-1234567890abcdef0
```

Contoh berikut mengubah target ke grup Reservasi Kapasitas tertentu.

```
Edit-EC2InstanceCapacityReservationAttribute `
    -InstanceId i-1234567890abcdef0 `
    -CapacityReservationTarget_CapacityReservationResourceGroupArn `
        "arn:aws:resource-groups:us-west-2:123456789012:group/my-cr-group"
```

Contoh berikut mengubah preferensi Reservasi Kapasitas menjadi`capacity-reservation-only`. Karena tidak menentukan Reservasi Kapasitas, instans diluncurkan ke Reservasi Kapasitas terbuka apa pun dengan atribut yang cocok dan kapasitas yang tersedia.

```
Edit-EC2InstanceCapacityReservationAttribute `
```

```
    -InstanceId i-1234567890abcdef0 `
    -CapacityReservationSpecification_CapacityReservationPreference "capacity-
reservation-only"
```

Contoh berikut mengubah preferensi Reservasi Kapasitas ke `capacity-reservation-only` dan mengubah target ke Reservasi Kapasitas tertentu. Jika kapasitas tidak tersedia di Reservasi Kapasitas yang ditentukan, instans gagal diluncurkan.

```
Edit-EC2InstanceCapacityReservationAttribute `
    -InstanceId i-1234567890abcdef0 `
    -CapacityReservationSpecification_CapacityReservationPreference "capacity-
reservation-only" `
    -CapacityReservationTarget_CapacityReservationId cr-1234567890abcdef0
```

### Memindahkan kapasitas antara Reservasi Kapasitas

Anda dapat memindahkan kapasitas dari satu Reservasi Kapasitas ke yang lain untuk mendistribusikan kembali sumber daya komputasi yang dipesan sesuai kebutuhan. Misalnya, jika Anda memerlukan kapasitas tambahan dalam reservasi dengan penggunaan yang terus meningkat, dan Anda memiliki kapasitas yang tersedia di reservasi lain, maka Anda dapat mengalokasikan kembali kapasitas tersebut di antara dua reservasi.

#### Prasyarat untuk memindahkan kapasitas

Sebagai prasyarat, dua Reservasi Kapasitas harus memenuhi persyaratan berikut:

- Kedua reservasi harus dalam keadaan aktif.
- Kedua reservasi harus dimiliki oleh Anda Akun AWS. Anda tidak dapat memindahkan kapasitas antara reservasi yang dimiliki oleh yang berbeda Akun AWS.
- Kedua reservasi harus memiliki hal yang sama:
  - Tipe instans
  - Platform
  - Zona Ketersediaan
  - Penghunian
  - Grup penempatan
  - Waktu akhir

Kelayakan instans Reservasi Kapasitas tujuan (`openataugmentargeted`), dan tag, tidak harus cocok dengan reservasi sumber. Konfigurasi kedua reservasi tetap sama, kecuali bahwa reservasi sumber telah mengurangi kapasitas dan reservasi tujuan telah meningkatkan kapasitas.

Bila Anda menentukan jumlah instans yang akan dipindahkan, secara default, kapasitas yang tersedia dipindahkan terlebih dahulu, diikuti oleh instans berjalan yang memenuhi syarat (kapasitas yang digunakan dalam reservasi Anda). Misalnya, jika Anda memindahkan 4 instans dari reservasi dengan 5 instans bekas dan 3 instans yang tersedia, maka 3 instans yang tersedia dan 1 instans yang digunakan akan dipindahkan.

> **Note**
>
> Saat memindahkan kapasitas bekas dari reservasi dengan menentukan  Jumlah yang akan dipindahkan  yang lebih besar dari kapasitas yang tersedia, hanya instans yang diluncurkan dengan Spesifikasi Reservasi  Kapasitas  as yang `open` akan dipindahkan.

Pertimbangan-pertimbangan

Pertimbangan berikut berlaku saat memindahkan kapasitas dari satu reservasi ke reservasi lainnya:

- Kapasitas yang digunakan hanya dapat dipindahkan antara Reservasi Kapasitas dengan kelayakan `open` instans yang dibagikan dengan kumpulan akun yang sama.
- Saat Anda memindahkan kapasitas yang digunakan, instans yang memenuhi syarat dipilih secara acak. Anda tidak dapat menentukan instans yang sedang berjalan yang dipindahkan. Jika jumlah instans yang memenuhi syarat tidak ditemukan untuk memenuhi kuantitas pemindahan, operasi pemindahan akan gagal.
- Jika Anda memindahkan semua kapasitas dari reservasi sumber, Reservasi Kapasitas akan dibatalkan secara otomatis.
- Future-dated Reservasi Kapasitas  — Anda tidak dapat memindahkan kapasitas untuk Reservasi Kapasitas yang akan datang selama periode komitmen.

> **Note**
>
> Memindahkan kapasitas dari Blok Kapasitas tidak didukung.

### Pindahkan kapasitas

Anda dapat memindahkan kapasitas dari Reservasi Kapasitas sumber ke Reservasi Kapasitas tujuan.

#### Console

**Untuk memindahkan kapasitas**

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Reservasi Kapasitas.
3. Pilih ID On-Demand Reservasi Kapasitas yang memiliki kapasitas untuk dipindahkan.
4. Di bawah Tindakan, Kelola kapasitas, pilih Pindahkan.
5. Pada halaman Pindahkan kapasitas, di bawah Reservasi Kapasitas Tujuan, pilih reservasi dari daftar.
6. Di bawah Jumlah yang akan dipindahkan, gunakan slider atau ketik jumlah instans untuk pindah dari sumber Reservasi Kapasitas ke Reservasi Kapasitas tujuan.
7. Tinjau ringkasan, dan ketika Anda siap, pilih Pindahkan.

#### AWS CLI

**Untuk memindahkan kapasitas**

Gunakan perintah move-capacity-reservation-instance. Contoh berikut memindahkan 10 instans dari sumber Reservasi Kapasitas yang ditentukan ke Reservasi Kapasitas tujuan yang ditentukan.

```
aws ec2 move-capacity-reservation-instances \
    --source-capacity-reservation-id cr-1234567890abcdef0 \
    --destination-capacity-reservation-id cr-021345abcdef56789 \
    --instance-count 10
```

#### PowerShell

**Untuk memindahkan kapasitas**

Gunakan Move-EC2CapacityReservationInstance cmdlet. Contoh berikut memindahkan 10 instans dari sumber Reservasi Kapasitas yang ditentukan ke Reservasi Kapasitas tujuan yang ditentukan.

```
Move-EC2CapacityReservationInstance `
    -SourceCapacityReservationId cr-1234567890abcdef0 `
    -DestinationCapacityReservationId cr-021345abcdef56789 `
    -InstanceCount 10
```

Memisahkan kapasitas dari Reservasi Kapasitas yang ada

Anda dapat memisahkan kapasitas dari Reservasi Kapasitas yang ada untuk membuat reservasi baru. Dengan membagi kapasitas, Anda mengalokasikan sebagian dari reservasi asli ke beban kerja tertentu atau membagikannya dengan yang lain Akun AWS. Misalnya, untuk berbagi sebagian Reservasi Kapasitas dengan akun lain, Anda dapat memisahkan sebagian kapasitas untuk membuat Reservasi Kapasitas berukuran lebih kecil. Reservasi Kapasitas berukuran lebih kecil kemudian dapat dibagikan dengan akun lain yang menggunakan AWS Resource Access Manager.

Saat Anda membagi kapasitas dari Reservasi Kapasitas yang ada, Reservasi Kapasitas baru secara otomatis dibuat. Reservasi yang ada tidak akan berubah, kecuali pengurangan kapasitas total dari jumlah instans yang dipisahkan. Instans yang berjalan di Reservasi Kapasitas yang ada tidak terpengaruh. Anda dapat membagi reservasi yang ada menjadi hanya satu Reservasi Kapasitas baru.

Reservasi Kapasitas baru akan memiliki konfigurasi yang sama dengan Reservasi Kapasitas yang ada kecuali untuk tag. Secara default, Reservasi Kapasitas baru tidak memiliki tag apa pun. Anda dapat menentukan tag baru selama operasi split. Reservasi Kapasitas baru juga dapat dimodifikasi setelah dibuat, jika perlu.

Bila Anda menentukan jumlah instans yang akan dibagi, secara default, kapasitas yang tersedia dibagi terlebih dahulu, diikuti oleh instans berjalan yang memenuhi syarat (kapasitas yang digunakan dalam reservasi Anda). Misalnya: jika Anda memisahkan 4 instans dari Reservasi Kapasitas dengan 5 instans bekas dan 3 instans yang tersedia, maka 3 instans yang tersedia dan 1 instans bekas akan dibagi menjadi reservasi baru.

Prasyarat untuk memisahkan kapasitas

Sebagai prasyarat, Reservasi Kapasitas Anda harus memenuhi persyaratan berikut:

- Reservasi sumber harus dalam keadaan aktif.
- Reservasi sumber harus dimiliki oleh Anda Akun AWS.

> **Note**
>
> Bila Anda membagi kapasitas yang digunakan dari reservasi dengan menentukan Ku antitas yang akan dibagi yang lebih besar dari kapasitas yang tersedia, hanya instans yang diluncurkan dengan Spesifikasi Reservasi Kapasitas as yang open akan dibagi.

Pertimbangan-pertimbangan

Pertimbangan berikut berlaku saat memisahkan kapasitas dari satu reservasi ke yang baru:

- Kapasitas yang digunakan hanya dapat dibagi untuk Reservasi Kapasitas dengan kelayakan instans “terbuka” yang tidak dibagikan dengan akun mana pun.
- Saat Anda membagi kapasitas yang digunakan, instans yang memenuhi syarat dipilih secara acak. Anda tidak dapat menentukan instans yang sedang berjalan yang dibagi. Jika jumlah instans yang memenuhi syarat tidak ditemukan untuk memenuhi kuantitas pemisahan, operasi pemisahan akan gagal.
- Jumlah maksimum instans yang akan dipisahkan dari reservasi yang ada adalah ukuran reservasi dikurangi satu. Misalnya, jika total kapasitas reservasi Anda adalah 5 instans, Anda dapat membagi maksimal 4 instans menjadi reservasi baru.
- Future-dated Reservasi Kapasitas — Anda tidak dapat membagi kapasitas untuk Reservasi Kapasitas di masa mendatang selama periode komitmen.
- Grup Sumber Daya Reservasi Kapasitas — Jika Reservasi Kapasitas yang ada milik Grup Sumber Daya Reservasi Kapasitas, Reservasi Kapasitas baru tidak akan ditambahkan secara otomatis ke Grup Sumber Daya Reservasi Kapasitas. Anda dapat menambahkan Reservasi Kapasitas baru ke Grup Sumber Daya Reservasi Kapasitas setelah dibuat, jika perlu.
- Ber bagi — Jika Reservasi Kapasitas yang ada dibagikan dengan akun konsumen, Reservasi Kapasitas baru tidak akan dibagikan secara otomatis dengan akun konsumen. Anda dapat membagikan Reservasi Kapasitas baru setelah dibuat, jika perlu.
- Grup penempatan klaster — Jika Reservasi Kapasitas yang ada merupakan bagian dari grup penempatan cluster, Reservasi Kapasitas baru akan dibuat dalam grup penempatan cluster yang sama.

> **Note**
>
> Memisahkan kapasitas dari Blok Kapasitas tidak didukung.

Kontrol akses untuk memisahkan Reservasi Kapasitas menggunakan tag

Anda dapat menggunakan tag untuk mengontrol akses ke sumber daya Amazon EC2, termasuk memisahkan kapasitas dari Reservasi Kapasitas yang ada untuk membuat Reservasi Kapasitas baru. Untuk informasi selengkapnya, lihat Mengontrol akses ke AWS sumber daya menggunakan tag di Panduan Pengguna IAM.

Untuk mengontrol akses untuk memisahkan Reservasi Kapasitas menggunakan tag, pastikan Anda menentukan tag sumber daya dan permintaan dalam pernyataan kebijakan karena kebijakan IAM dievaluasi terhadap Reservasi Kapasitas sumber dan Reservasi Kapasitas yang baru dibuat. Kebijakan contoh berikut menyertakan kunci `ec2:ResourceTag` kondisi dengan tag `Owner=ExampleDepartment1` untuk Reservasi Kapasitas sumber dan kunci `ec2:RequestTag` kondisi dengan tag `stack=production` untuk Reservasi Kapasitas yang baru dibuat.

```
{
  "Statement": [
    {
      "Sid": "AllowSourceCapacityReservation",
      "Effect": "Allow",
      "Action": "ec2:CreateCapacityReservationBySplitting",
      "Resource": "arn:aws:ec2:us-east-1:111122223333:capacity-reservation/
cr-1234567890abcdef0",
      "Condition": {
        "StringEquals": {
          "ec2:ResourceTag/Owner": "ExampleDepartment1"
        }
      }
    },
    {
      "Sid": "AllowNewlyCreatedCapacityReservation",
      "Effect": "Allow",
      "Action": ["ec2:CreateCapacityReservationBySplitting", "ec2:CreateTags"],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:capacity-reservation/*",
      "Condition": {
        "StringEquals": {
```

```
          "aws:RequestTag/stack": "production"
        }
      }
    }
  ]
}
```

Pisahkan kapasitas

Anda dapat memisahkan kapasitas dari Reservasi Kapasitas yang ada untuk membuat Reservasi Kapasitas baru.

Console

Untuk memisahkan kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Reservasi Kapasitas.
3. Pilih ID On-Demand Reservasi Kapasitas yang memiliki kapasitas untuk dibagi.
4. Di bawah T indakan, Kelola kapasitas, pilih Pisahkan.
5. Pada halaman Pem bagian Kapasitas Reservasi, di bawah Jumlah yang akan dibagi, gunakan slider atau ketik jumlah instans yang akan dipisahkan dari reservasi saat ini.
6. (Opsional) Tambahkan tag untuk Reservasi Kapasitas baru.
7. Tinjau ringkasan, dan ketika Anda siap, pilih Pis ahkan.

AWS CLI

Untuk memisahkan kapasitas

Gunakan perintah `create-capacity-reservation-by-splitting`. Contoh berikut membuat Reservasi Kapasitas baru dengan memisahkan 10 instans dari Reservasi Kapasitas yang ditentukan.

```
aws ec2 create-capacity-reservation-by-splitting \
    --source-capacity-reservation-id cr-1234567890abdef0 \
    --instance-count 10
```

PowerShell

Untuk memisahkan kapasitas

Gunakan New-EC2CapacityReservationBySplitting cmdlet. Contoh berikut membuat Reservasi Kapasitas baru dengan memisahkan 10 instans dari Reservasi Kapasitas yang ditentukan.

```
New-EC2CapacityReservationBySplitting `
    -SourceCapacityReservationId cr-1234567890abdef0 `
    -InstanceCount 10
```

Membatalkan Reservasi Kapasitas

Anda dapat membatalkan Reservasi Kapasitas yang berada di salah satu status berikut:

- `assessing`
- `scheduled`
- `active`
- `delayed`

> **Note**
>
> Anda tidak dapat memodifikasi atau membatalkan Blok Kapasitas. Untuk informasi selengkapnya, lihat Blok Kapasitas untuk ML.

Jika Reservasi Kapasitas tanpa `active` durasi komitmen atau durasi komitmen yang telah berlalu, pembatalan gratis.

Jika Reservasi Kapasitas adalah `assessing` ataudelayed, pembatalan gratis.

Jika Reservasi Kapasitas adalah `scheduled` atau `active` dengan durasi komitmen yang tersisa, biaya pembatalan mungkin berlaku. Untuk informasi selengkapnya, lihat Biaya pembatalan.

Jika Anda membatalkan Reservasi Kapasitas yang memiliki instans yang sedang berjalan, instans akan terus berjalan secara normal di luar Reservasi Kapasitas dengan tarif On-Demand Instans standar atau dengan tarif diskon jika Anda memiliki Paket Tab ungan atau Instans Cadangan Regional yang sesuai.

Setelah Anda membatalkan Reservasi Kapasitas, instans yang menargetkannya tidak dapat diluncurkan lagi. Modifikasi instans ini sehingga mereka menargetkan Reservasi Kapasitas yang berbeda, meluncurkan Reservasi Kapasitas terbuka dengan atribut yang cocok dan kapasitas yang memadai, atau menghindari peluncuran ke Reservasi Kapasitas. Untuk informasi selengkapnya, lihat Mengubah setelan Reservasi Kapasitas instans Anda.

Topik

- Biaya pembatalan
- Penawaran pembatalan
- Membatalkan Reservasi Kapasitas

## Biaya pembatalan

Jika Anda membatalkan Reservasi Kapasitas di masa mendatang dalam waktu 8 minggu sejak tanggal mulai, atau selama durasi komitmen setelah pengiriman, Anda akan dikenakan biaya persentase dari durasi komitmen Anda. Kapasitas dilepaskan segera setelah Anda membatalkan, tetapi reservasi terus ditagih dengan On-Demand tarif hingga akhir periode komitmen yang dikurangi. Jika Anda memiliki Paket Tab ungan atau Instans Cad angan Regional yang sesuai, tarif diskon berlaku secara otomatis.

Untuk menghitung biaya, Amazon EC2 mengalikan durasi komitmen Anda dengan tingkat pembatalan. Semakin awal Anda membatalkan relatif terhadap tanggal mulai, semakin rendah tarifnya. Tabel berikut menunjukkan tingkat pembatalan.

Tarif pembatalan

| Saat Anda membatalkan | Tingkat pembatalan minimum |
|---|---|
| 8 minggu atau lebih sebelum tanggal mulai asli | Tidak dikenai biaya |
| 2 hingga 8 minggu sebelum tanggal mulai asli | 25% dari total komitmen |
| Dalam waktu 2 minggu dari tanggal mulai asli | 50% dari total komitmen |
| Setelah melahirkan, selama durasi komitmen | 50% dari komitmen yang tersisa |

Tabel ini memberikan panduan umum tentang biaya pembatalan. Penawaran pembatalan menentukan durasi biaya pembatalan yang tepat.

Misalnya, jika Anda memiliki Reservasi `c8g.xlarge` Kapasitas 70 instans dengan durasi komitmen 14 hari dan Anda membatalkan dalam waktu 2 minggu dari tanggal mulai awal (tarif pembatalan 50%), Anda akan dikenakan biaya komitmen selama 7 hari. Selama 7 hari itu, Anda membayar 70 instans dengan On-Demand tarif tersebut.

Ketika Anda dikenakan biaya untuk pembatalan:

- Reservasi Kapasitas memasuki `cancelling` negara bagian.
- Reservasi Kapasitas tidak dapat diubah, dipindahkan, dibagikan, atau dibagi.
- Reservasi Kapasitas tetap terlihat di konsol dan dalam `DescribeCapacityReservations` tanggapan.
- Transisi Reservasi Kapasitas ke `cancelled` saat komitmen berakhir.

Untuk mengetahui kapan biaya pembatalan berakhir, gunakan perintah describe-capacity-reservations dan periksa `CommitmentInfo.CommitmentEndDate` bidang dalam tanggapan. Biaya berhenti ketika komitmen berakhir dan Reservasi Kapasitas beralih ke`cancelled`. Untuk informasi tentang bagaimana biaya pembatalan muncul di AWS Laporan Biaya dan Penggunaan 2.0, lihatBiaya pembatalan di Laporan Biaya dan Penggunaan 2.0.

Jika Reservasi Kapasitas memiliki sisa durasi komitmen, Anda tidak dapat membatalkannya saat instans masih berjalan. Anda harus terlebih dahulu menghentikan atau menghentikan instans yang sedang berjalan, yang mencegah Anda dikenakan biaya pembatalan dan biaya On-Demand Instans secara bersamaan.

Untuk membatalkan Reservasi Kapasitas dengan biaya pembatalan, Anda harus terlebih dahulu membuat penawaran pembatalan dan menerima persyaratan. Untuk informasi selengkapnya, lihat Penawaran pembatalan.

Penawaran pembatalan

Jika pembatalan memerlukan biaya, Anda harus meninjau dan menerima penawaran pembatalan sebelum melanjutkan. Penawaran memberikan ketentuan pembatalan yang tepat, termasuk rincian berikut:

- Konfigurasi Reservasi Kapasitas saat ini
- Status yang dialihkan ke Reservasi Kapasitas setelah pembatalan (`cancelling`)
- Jumlah instans yang dikomit yang dikenakan biaya
- Durasi pengisian

- Tanggal akhir tagihan (saat komitmen berakhir dan Reservasi Kapasitas beralih ke`cancelled`)

Setelah Anda meninjau persyaratan, Anda mengkonfirmasi pembatalan dengan menerima penawaran.

Di konsol Amazon EC2, penawaran dibuat secara otomatis saat Anda memulai pembatalan. Anda dapat meninjau persyaratan dan mengonfirmasi penerimaan Anda secara langsung. Dengan API AWS CLI or, gunakan `CreateCapacityReservationCancellationQuote` untuk menghasilkan penawaran, lalu berikan ID kutipan ke`CancelCapacityReservation`.

Penawaran pembatalan berlaku selama 24 jam. Setelah 24 jam, Anda harus membuat penawaran baru. Untuk melihat penawaran pembatalan Anda yang ada, gunakan`DescribeCapacityReservationCancellationQuotes`.

> **Important**
>
> Pembatalan tidak dapat diubah. Setelah menerima penawaran pembatalan dan membatalkan Reservasi Kapasitas, Anda tidak dapat membatalkan pembatalan.

### Membatalkan Reservasi Kapasitas

Gunakan prosedur berikut untuk membatalkan Reservasi Kapasitas.

#### Console

**Untuk membatalkan Reservasi Kapasitas**

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Reservasi Kapasitas dan pilih Rserrvasi Kapasitas untuk dibatalkan.
3. Pilih Bat alkan reservasi.
4. Jika penawaran pembatalan diperlukan, tinjau ketentuan pembatalan, termasuk durasi tagihan dan tanggal akhir tagihan. Mas **`confirm`** uk untuk menerima persyaratan.
5. Pilih Bat alkan reservasi.

#### AWS CLI

**Untuk membatalkan Reservasi Kapasitas**

Gunakan perintah cancel-capacity-reservation.

```
aws ec2 cancel-capacity-reservation \
    --capacity-reservation-id cr-1234567890abcdef0
```

Jika penawaran pembatalan diperlukan, pertama-tama buat penawaran menggunakan create-capacity-reservation-cancellation-quote, lalu berikan ID penawaran ke perintah batal.

```
aws ec2 create-capacity-reservation-cancellation-quote \
    --capacity-reservation-id cr-1234567890abcdef0
```

Tinjau `CancellationTerms` tanggapan, termasuk `ChargeCommitmentDurationHours` dan`ChargeEndDate`. Kemudian batalkan dengan ID penawaran:

```
aws ec2 cancel-capacity-reservation \
    --capacity-reservation-id cr-1234567890abcdef0 \
    --quote-id crcq-1a2b3c4d5e6f7g8h9 \
    --apply-cancellation-charges commitment-wind-down
```

PowerShell

Untuk membatalkan Reservasi Kapasitas

Gunakan Remove-EC2CapacityReservation cmdlet.

```
Remove-EC2CapacityReservation `
    -CapacityReservationId cr-1234567890abcdef0
```

Jika penawaran pembatalan diperlukan, pertama-tama buat penawaran menggunakan New-EC2CapacityReservationCancellationQuote, lalu berikan ID penawaran ke cmdlet batal.

```
New-EC2CapacityReservationCancellationQuote `
    -CapacityReservationId cr-1234567890abcdef0
```

Tinjau ketentuan pembatalan dalam tanggapan. Kemudian batalkan dengan ID penawaran:

```
Remove-EC2CapacityReservation `
    -CapacityReservationId cr-1234567890abcdef0 `
```

```
    -QuoteId crcq-1a2b3c4d5e6f7g8h9 `
    -ApplyCancellationCharge "commitment-wind-down"
```

Gunakan Reservasi Kapasitas dengan grup penempatan

Anda dapat membuat Reservasi Kapasitas dalam grup penempatan untuk memesan kapasitas komputasi Amazon EC2 untuk beban kerja Anda.

Membuat Reservasi Kapasitas dalam grup penempatan memastikan bahwa Anda memiliki akses ke kapasitas komputasi di grup penempatan Anda saat Anda membutuhkannya, selama Anda membutuhkannya. Ini memungkinkan Anda untuk mengurangi skala sambil memastikan bahwa kapasitas tetap tersedia untuk Anda gunakan sehingga Anda dapat meningkatkan kembali saat diperlukan.

Setelah membuat Reservasi Kapasitas di grup penempatan, Anda dapat membagikannya dengan AWS akun lain. Untuk informasi selengkapnya, lihat Berbagi Kapasitas Reservasi dalam kelompok penempatan.

Strategi penempatan yang didukung

Anda dapat membuat Reservasi Kapasitas di grup penempatan yang menggunakan strategi berikut:

- Kluster
- Waktu presisi

> **Note**
>
> Grup penempatan Spread dan Partisi tidak mendukung Reservasi Kapasitas. Blok Kapasitas tidak mendukung grup penempatan.

Untuk informasi tentang jenis instans yang didukung, Wilayah yang tersedia, dan strategi penempatan, lihatGrup penempatan untuk instans Amazon EC2 Anda.

Topik

- Batasan
- Bekerja dengan Reservasi Kapasitas di grup penempatan

- Berbagi Kapasitas Reservasi dalam kelompok penempatan

### Batasan

Ingatlah hal-hal berikut saat membuat Reservasi Kapasitas di grup penempatan:

- Jika Reservasi Kapasitas yang ada tidak ada dalam grup penempatan, Anda tidak dapat mengubah Reservasi Kapasitas untuk memesan kapasitas dalam grup penempatan. Untuk reservasi kapasitas dalam grup penempatan, Anda harus membuat Reservasi Kapasitas di grup penempatan.
- Setelah membuat Reservasi Kapasitas di grup penempatan, Anda tidak dapat mengubahnya untuk memesan kapasitas di luar grup penempatan.
- Anda dapat meningkatkan kapasitas terpesan Anda dalam grup penempatan dengan memodifikasi Reservasi Kapasitas yang ada di grup penempatan, atau dengan membuat Reservasi Kapasitas tambahan di grup penempatan. Namun, Anda meningkatkan peluang Anda untuk mendapatkan kesalahan kapasitas yang tidak mencukupi.
- Anda dapat membagikan Reservasi Kapasitas hanya dari grup penempatan yang Anda miliki. Anda tidak dapat membagikan Reservasi Kapasitas dari grup penempatan yang tidak Anda miliki.
- Anda tidak dapat menghapus grup penempatan yang memiliki Reservasi `active` Kapasitas. Anda harus membatalkan semua Reservasi Kapasitas di grup penempatan sebelum Anda dapat menghapusnya.

### Bekerja dengan Reservasi Kapasitas di grup penempatan

Untuk mulai menggunakan Reservasi Kapasitas dengan grup penempatan, lakukan langkah-langkah berikut.

> **Note**
>
> Jika Anda ingin membuat Reservasi Kapasitas di grup penempatan yang ada, lewati Langkah 1. Kemudian untuk Langkah 2 dan 3, tentukan ARN dari grup penempatan yang ada.

### Tugas

- Langkah 1: (Bersyarat) Buat grup penempatan untuk digunakan dengan Reservasi Kapasitas
- Langkah 2: Buat Reservasi Kapasitas di grup penempatan

- Langkah 3: Luncurkan instans ke Reservasi Kapasitas di grup penempatan

Langkah 1: (Bersyarat) Buat grup penempatan untuk digunakan dengan Reservasi Kapasitas

Lakukan langkah ini hanya jika Anda perlu membuat grup penempatan baru. Untuk menggunakan grup penempatan yang ada, lewati langkah ini dan kemudian untuk Langkah 2 dan 3, gunakan ARN grup penempatan tersebut.

Console

Untuk membuat grup penempatan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Grup Penempatan, lalu pilih Buat grup penempatan.
3. Untuk Nama, tentukan nama deskriptif untuk grup penempatan.
4. Untuk Strategi Pen empatan, pilih strategi penempatan.
5. Pilih Buat grup.
6. Di tabel Grup Pen empatan, di kolom Grup ARN, buat catatan ARN dari grup penempatan yang Anda buat. Anda akan membutuhkannya untuk langkah selanjutnya.

AWS CLI

Untuk membuat grup penempatan

Gunakan perintah create-placement-group.

```
aws ec2 create-placement-group \
    --group-name MyPG \
    --strategy strategy
```

Catat grup penempatan ARN yang dikembalikan dalam output, karena Anda akan membutuhkannya untuk langkah berikutnya.

PowerShell

Untuk membuat grup penempatan

Gunakan New-EC2PlacementGroup cmdlet.

```
New-EC2PlacementGroup `
    -GroupName my-placement-group `
    -Strategy strategy
```

Catat grup penempatan ARN yang dikembalikan dalam output, karena Anda akan membutuhkannya untuk langkah berikutnya.

Langkah 2: Buat Reservasi Kapasitas di grup penempatan

Anda membuat Reservasi Kapasitas di grup penempatan dengan cara yang sama seperti Anda membuat Reservasi Kapasitas apa pun. Namun, Anda juga harus menentukan ARN grup penempatan tempat membuat Reservasi Kapasitas.

Pertimbangan-pertimbangan

- Grup penempatan yang ditentukan harus berada di `available` negara bagian. Jika grup penempatan berada di `pending` status`deleting`,, atau`deleted`, permintaan gagal.
- Reservasi Kapasitas dan grup penempatan harus berada di Zona Ketersediaan yang sama. Jika permintaan untuk membuat Reservasi Kapasitas menentukan Zona Ketersediaan yang berbeda dari grup penempatan, permintaan gagal.
- Anda dapat membuat Reservasi Kapasitas hanya untuk jenis contoh yang didukung oleh grup penempatan. Jika Anda menentukan tipe instans yang tidak didukung, permintaan gagal.
- Jika Anda membuat Reservasi `open` Kapasitas di grup penempatan dan ada instans yang sedang berjalan yang memiliki atribut yang cocok (grup penempatan ARN, jenis instans, Zona Ketersediaan, platform, dan sewa), instans tersebut secara otomatis berjalan di Reservasi Kapasitas.
- Permintaan Anda untuk membuat Reservasi Kapasitas bisa gagal jika salah satu dari yang berikut ini benar:
  - Amazon EC2 tidak memiliki kapasitas yang cukup untuk memenuhi permintaan. Coba lagi nanti, coba Zona Ketersediaan yang berbeda, atau coba kapasitas yang lebih kecil. Jika beban kerja Anda fleksibel di semua tipe dan ukuran instans, coba atribut instans yang berbeda.
  - Kuantitas yang diminta melebihi batas On-Demand Instans Anda untuk keluarga instans yang dipilih. Tingkatkan batas On-Demand Instans untuk keluarga instans dan coba lagi. Untuk informasi selengkapnya, lihat Kuota Instans Sesuai Permintaan.

Console

Untuk membuat Reservasi Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Reservasi Kapasitas, lalu pilih Buat Reservasi Kapasitas.
3. Pada halaman Buat Reservasi Kapasitas, tentukan jenis instans, platform, Zona Ketersediaan, Sewa, kuantitas, dan tanggal akhir sesuai kebutuhan.
4. Untuk Grup Penempatan, pilih ARN dari grup penempatan tempat untuk membuat Reservasi Kapasitas.
5. Pilih Buat.

Untuk informasi selengkapnya, lihat Membuat Reservasi Kapasitas.

AWS CLI

Untuk membuat Reservasi Kapasitas

Gunakan perintah create-capacity-reservation. Untuk `--placement-group-arn`, tentukan ARN grup penempatan tempat membuat Reservasi Kapasitas.

```
aws ec2 create-capacity-reservation \
    --instance-type instance_type \
    --instance-platform platform \
    --availability-zone-id az_id \
    --instance-count quantity \
    --placement-group-arn "placement_group_arn"
```

PowerShell

Untuk membuat Reservasi Kapasitas

Gunakan Add-EC2CapacityReservation cmdlet. Untuk `-PlacementGroupArn`, tentukan ARN grup penempatan tempat membuat Reservasi Kapasitas.

```
Add-EC2CapacityReservation `
    -InstanceType instance_type `
    -InstancePlatform platform `
    -AvailabilityZoneId az_id `
    -InstanceCount quantity `
```

```
    -PlacementGroupArn "placement_group_arn"
```

Langkah 3: Luncurkan instans ke Reservasi Kapasitas di grup penempatan

Anda dapat meluncurkan instance ke Reservasi Kapasitas yang ada dalam grup penempatan dengan salah satu opsi berikut:

- Menentukan ARN grup penempatan tempat peluncuran instans — Saat Anda memberikan ARN grup penempatan, Amazon EC2 meluncurkan instans ke grup penempatan tersebut. Gunakan salah satu metode berikut:
  - Men *open* entukan — Anda tidak perlu menentukan Reservasi Kapasitas dalam permintaan peluncuran instans. Jika instans memiliki atribut (grup penempatan ARN, tipe instans, Zona Ketersediaan, platform, dan penghunian) yang cocok dengan Reservasi Kapasitas dalam grup penempatan yang ditentukan, instans secara otomatis berjalan di Reservasi Kapasitas.
  - Menentukan Reservasi Kapasitas — Jika Reservasi Kapasitas hanya menerima peluncuran instans yang ditargetkan, Anda harus menentukan Reservasi Kapasitas target selain grup penempatan dalam permintaan.
  - Menentukan grup Reservasi Kapasitas — Untuk informasi selengkapnya, lihat Menggunakan Reservasi Kapasitas di grup penempatan dengan grup Reservasi Kapasitas.
- Menentukan hanya grup Reservasi Kapasitas — Untuk informasi selengkapnya, lihat Menggunakan Reservasi Kapasitas di grup penempatan dengan grup Reservasi Kapasitas.
- Menentukan hanya Reservasi Kapasitas — Anda dapat meluncurkan instans ke Reservasi Kapasitas dalam grup penempatan.

> **Note**
>
> Saat Anda meluncurkan instans dengan hanya menentukan Reservasi Kapasitas atau hanya grup Reservasi Kapasitas, instans diluncurkan ke Reservasi Kapasitas yang dibuat di grup penempatan, tetapi instans tidak langsung dilampirkan ke grup penempatan.

Console

Untuk meluncurkan instans ke Reservasi Kapasitas yang ada

1. Ikuti prosedur untuk meluncurkan instance, tetapi jangan meluncurkan instans sampai Anda menyelesaikan langkah-langkah berikut untuk menentukan pengaturan untuk grup penempatan dan Reservasi Kapasitas.
2. Per luas detail lanj utan dan lakukan hal berikut:
   a. Untuk Grup Pen empatan, pilih grup penempatan tempat untuk meluncurkan instance.
   b. Untuk Reservasi Kapasitas, pilih salah satu opsi berikut, tergantung pada konfigurasi Reservasi Kapasitas:
      - Buka — Untuk meluncurkan instans ke Reservasi `open` Kapasitas apa pun di grup penempatan yang memiliki atribut yang cocok dan kapasitas yang memadai.
      - Targetkan berdasarkan ID — Untuk meluncurkan instans ke Reservasi Kapasitas yang hanya menerima peluncuran instans yang ditargetkan.
      - Target berdasarkan grup — Untuk meluncurkan instans ke Reservasi Kapasitas apa pun dengan atribut yang cocok dan kapasitas yang tersedia di grup Reservasi Kapasitas yang dipilih.
3. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

AWS CLI

Untuk meluncurkan instans ke Reservasi Kapasitas yang ada

Gunakan perintah run-instans. Jika Anda perlu menargetkan Reservasi Kapasitas atau grup Reservasi Kapasitas tertentu, tentukan parameter `--capacity-reservation-specification`. Untuk `--placement`, tentukan parameter `GroupName` lalu tentukan nama grup penempatan yang Anda buat di langkah sebelumnya.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count quantity \
    --instance-type instance_type \
    --key-name key_pair_name \
```

```
    --subnet-id subnet-0abcdef1234567890 \
    --capacity-reservation-specification
 CapacityReservationTarget={CapacityReservationId=capacity_reservation_id} \
    --placement "GroupName=placement_group_name"
```

PowerShell

Untuk meluncurkan instans ke Reservasi Kapasitas yang ada

Gunakan New-EC2Instance cmdlet. Untuk `-Placement`, tentukan parameter `GroupName` lalu tentukan nama grup penempatan yang Anda buat di langkah sebelumnya.

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType instance_type `
    -KeyName key_pair_name `
    -SubnetId subnet-0abcdef1234567890 `
    -CapacityReservationTarget_CapacityReservationId capacity_reservation_id `
    -Placement_GroupName placement_group_name
```

Berbagi Kapasitas Reservasi dalam kelompok penempatan

Anda dapat membagikan Reservasi Kapasitas dalam grup penempatan dengan hanya membagikan Reservasi Kapasitas, atau dengan membagikan Reservasi Kapasitas dan grup penempatan tempat reservasi tersebut dibuat.

Dengan hanya membagikan Reservasi Kapasitas, Anda memberikan akses akun konsumen ke Reservasi Kapasitas itu saja. Akun konsumen tidak memiliki visibilitas atau akses ke grup penempatan tempat Reservasi Kapasitas dibuat. Ini memberi Anda kontrol halus atas akses akun konsumen. Akun konsumen tidak dapat melihat informasi apa pun tentang grup penempatan, termasuk ARN-nya.

Saat Anda membagikan grup penempatan dan Reservasi Kapasitas, grup penempatan terlihat dan dapat diakses oleh akun konsumen. Mereka dapat meluncurkan instance dan membuat Reservasi Kapasitas mereka sendiri di dalamnya.

Untuk informasi selengkapnya, lihat sumber daya berikut ini.

- Meluncurkan instans ke Reservasi Kapasitas dalam grup penempatan
- Reservasi Kapasitas Bersama

- Grup penempatan bersama

### Reservasi Kapasitas di Local Zones

Zona Lokal adalah perpanjangan dari Wil AWS ayah yang secara geografis dekat dengan pengguna Anda. Sumber daya yang dibuat di Zona Lokal dapat melayani pengguna lokal dengan komunikasi latensi sangat rendah. Untuk informasi selengkapnya, lihat AWS Local Zones.

Anda dapat memperluas VPC dari Wil AWS ayah induknya ke Zona Lokal dengan membuat subnet baru di Zona Lokal tersebut. Saat Anda membuat subnet di Zona Lokal, VPC Anda diperluas ke Zona Lokal itu. Subnet di Zona Lokal beroperasi sama dengan subnet lain di VPC Anda.

Dengan Local Zones, Anda dapat menempatkan Reservasi Kapasitas di banyak lokasi yang lebih dekat dengan pengguna Anda. Anda membuat dan menggunakan Reservasi Kapasitas di Local Zones dengan cara yang sama seperti Anda membuat dan menggunakan Reservasi Kapasitas di Zona Ketersediaan biasa. Fitur yang sama dan perilaku pencocokan instans yang berlaku. Untuk informasi selengkapnya tentang model harga yang didukung di Local Zones, lihat FAQ AWS Local Zones.

#### Pertimbangan-pertimbangan

Anda tidak dapat menggunakan grup Reservasi Kapasitas dalam Zona Lokal.

#### Untuk menggunakan Reservasi Kapasitas di Zona Lokal

1. Aktifkan Zona Lokal untuk digunakan di AWS akun Anda. Untuk informasi selengkapnya, lihat Mem ulai dengan AWS Zona Lok AWS al di Panduan Pengguna Zona Lokal.
2. Buat Reservasi Kapasitas di Zona Lokal. Untuk Zona Ketersediaan, pilih Zona Lokal. Zona Lokal ditunjukkan oleh kode Wilayah AWS dan diikuti oleh pengidentifikasi yang menunjukkan lokasinya, misalnya `us-west-2-lax-1a`. Untuk informasi selengkapnya, lihat Membuat Reservasi Kapasitas.
3. Buat subnet di Zona Lokal. Untuk Zona Ketersediaan, pilih Zona Lokal. Untuk informasi selengkapnya, lihat Membuat subnet di VPC Anda dalam Panduan Pengguna Amazon VPC.
4. Luncurkan sebuah instans. Untuk Subnet, pilih subnet di Zona Lokal (misalnya `subnet-123abc | us-west-2-lax-1a`), dan untuk Reservasi Kapasitas, pilih spesifikasi (baik `open` atau targetkan menurut ID) yang diperlukan untuk Reservasi Kapasitas yang Anda buat di Zona Lokal. Untuk informasi selengkapnya, lihat Luncurkan instans ke dalam Reservasi Kapasitas yang ada.

### Reservasi Kapasitas di Wavelength Zones

AWS Wavelength memungkinkan developer untuk membangun aplikasi yang menghasilkan latensi sangat rendah untuk perangkat seluler dan pengguna akhir. Wavelength melakukan deployment layanan komputasi dan penyimpanan AWS standar ke edge jaringan 5G operator telekomunikasi. Anda dapat memperluas Amazon Virtual Private Cloud (VPC) ke satu atau beberapa Wavelength Zones. Anda kemudian dapat menggunakan AWS sumber daya seperti instans Amazon EC2 untuk menjalankan aplikasi yang memerlukan latensi sangat rendah dan koneksi ke AWS layanan di Wilayah. Untuk informasi selengkapnya, lihat AWS Wavelength Zones.

Saat Anda membuat Reservasi On-Demand Kapasitas, Anda dapat memilih Zona Panjang Gelombang dan Anda dapat meluncurkan instance ke Reservasi Kapasitas di Zona Panjang Gelombang dengan menentukan subnet yang terkait dengan Zona Panjang Gelombang. Zona Panjang Gelombang diwakili oleh kode Wil AWS ayah diikuti oleh pengenal yang menunjukkan lokasi, misalnya`us-east-1-wl1-bos-wlz-1`.

Wavelength Zones tidak tersedia di setiap Wilayah. Untuk informasi tentang Wilayah yang mendukung Wavelength Zones, lihat Wavelength Zones yang Tersedia di Panduan Developer AWS Wavelength .

#### Pertimbangan-pertimbangan

Anda tidak dapat menggunakan grup Reservasi Kapasitas dalam Zona Wavelength.

#### Untuk menggunakan Reservasi Kapasitas di Zona Wavelength

1. Aktifkan Zona Panjang Gelombang untuk digunakan di AWS akun Anda. Untuk informasi selengkapnya, lihat Memulai dengan AWS Wavelength di Panduan Developer AWS Wavelength .
2. Buat Reservasi Kapasitas di Zona Wavelength. Untuk Zona Ketersediaan, pilih Wavelength. Wavelength ditunjukkan oleh kode Wilayah AWS dan diikuti oleh pengidentifikasi yang menunjukkan lokasinya, misalnya `us-east-1-wl1-bos-wlz-1`. Untuk informasi selengkapnya, lihat Membuat Reservasi Kapasitas.
3. Buat subnet di Zona Wavelength. Untuk Zona Ketersediaan, pilih Zona Wavelength. Untuk informasi selengkapnya, lihat Membuat subnet di VPC Anda dalam Panduan Pengguna Amazon VPC.
4. Luncurkan sebuah instans. Untuk Subnet, pilih subnet di Wavelength Zone (misalnya `subnet-123abc | us-east-1-wl1-bos-wlz-1`), dan untuk Reservasi Kapasitas, pilih spesifikasi (baik `open` atau targetkan menurut ID) yang diperlukan untuk reservasi Kapasitas

yang Anda buat di Wavelength. Untuk informasi selengkapnya, lihat Luncurkan instans ke dalam Reservasi Kapasitas yang ada.

### Pencadangan Kapasitas di AWS Outposts

AWS Outposts adalah layanan yang dikelola sepenuhnya yang memperluas AWS infrastruktur, layanan, API, dan alat ke tempat pelanggan. Dengan menyediakan akses lokal ke infrastruktur AWS terkelola, AWS Outposts memungkinkan pelanggan untuk membangun dan menjalankan aplikasi di tempat menggunakan antarmuka pemrograman yang sama seperti di Wil AWS ayah, sambil menggunakan sumber daya komputasi dan penyimpanan lokal untuk latensi yang lebih rendah dan kebutuhan pemrosesan data lokal.

Outpost adalah kumpulan kapasitas AWS komputasi dan penyimpanan yang digunakan di situs pelanggan. AWS mengoperasikan, memantau, dan mengelola kapasitas ini sebagai bagian dari Wil AWS ayah.

Anda dapat membuat Reservasi Kapasitas pada Outposts yang telah Anda buat di akun Anda. Hal ini memungkinkan Anda untuk memesan kapasitas komputasi pada Outpost di situs Anda. Anda membuat dan menggunakan Reservasi Kapasitas di Outposts dengan cara yang sama seperti Anda membuat dan menggunakan Reservasi Kapasitas di Zona Ketersediaan biasa. Fitur yang sama dan perilaku pencocokan instans yang berlaku.

Anda juga dapat membagikan Reservasi Kapasitas di Outpost dengan AWS akun lain dalam organisasi Anda menggunakan AWS Resource Access Manager. Untuk informasi selengkapnya tentang berbagi Reservasi Kapasitas, lihat Reservasi Kapasitas Bersama.

#### Prasyarat

Anda harus menginstal Outpost di situs Anda. Untuk informasi selengkapnya, lihat Membuat Outpost dan memesan kapasitas Outpost di Panduan Pengguna AWS Outposts .

#### Pertimbangan-pertimbangan

- Anda tidak dapat menggunakan grup Reservasi Kapasitas di Outpost.

#### Untuk menggunakan Reservasi Kapasitas di Outpost

1. Buat subnet pada Outpost. Untuk informasi selengkapnya, lihat Membuat subnet di Panduan Pengguna AWS Outposts .
2. Buat Reservasi Kapasitas di Outpost.

a. Buka AWS Outposts konsol di https://console.aws.amazon.com/outposts/.

b. Di panel navigasi, pilih Outposts, lalu pilih Tindakan, Buat Reservasi Kapasitas.

c. Konfigurasikan Reservasi Kapasitas sesuai kebutuhan kemudian pilih Buat. Untuk informasi selengkapnya, lihat Membuat Reservasi Kapasitas.

> **Note**
>
> Daftar tarik-turun Tipe Instans hanya mencantumkan tipe instans yang didukung oleh Outpost yang dipilih, dan tarik-turun Zona Ketersediaan hanya mencantumkan Zona Ketersediaan yang terkait dengan Outpost yang dipilih.

3. Luncurkan sebuah instans ke dalam Reservasi Kapasitas. Untuk Subnet, pilih subnet yang Anda buat di Langkah 1, dan untuk Reservasi Kapasitas, pilih Reservasi Kapasitas yang Anda buat pada Langkah 2. Untuk informasi selengkapnya, lihat Meluncurkan instans pada Outpost di Panduan Pengguna AWS Outposts .

### Reservasi Kapasitas Bersama

Pembagian Reservasi Kapasitas memungkinkan pemilik Reservasi Kapasitas berbagi kapasitas yang dipesan dengan AWS akun lain atau dalam AWS organisasi. Ini memungkinkan Anda untuk membuat dan mengelola Reservasi Kapasitas secara terpusat, dan berbagi kapasitas yang dipesan di beberapa AWS akun atau di dalam AWS organisasi Anda.

Dalam model ini, AWS akun yang memiliki Reservasi Kapasitas (pemilik) membagikannya dengan AWS akun lain (konsumen). Konsumen dapat meluncurkan instans ke Reservasi Kapasitas yang dibagikan dengan mereka dengan cara yang sama seperti mereka meluncurkan instans ke Reservasi Kapasitas yang mereka miliki di akun mereka sendiri. Pemilik Reservasi Kapasitas bertanggung jawab untuk mengelola Reservasi Kapasitas dan instans yang diluncurkan ke dalamnya. Pemilik tidak dapat memodifikasi instans yang diluncurkan konsumen ke Reservasi Kapasitas yang telah mereka bagikan. Konsumen bertanggung jawab untuk mengelola instans yang mereka luncurkan ke Reservasi Kapasitas yang dibagikan dengan mereka. Konsumen tidak dapat menampilkan atau memodifikasi instans yang dimiliki oleh konsumen lain atau oleh pemilik Reservasi Kapasitas.

Pemilik Reservasi Kapasitas dapat berbagi Reservasi Kapasitas dengan:

- AWS Akun tertentu di dalam atau di luar AWS organisasinya
- Unit organisasi di dalam AWS organisasinya

- Seluruh AWS organisasinya

Prasyarat untuk berbagi Reservasi Kapasitas

- Untuk berbagi Reservasi Kapasitas, Anda harus memilikinya di AWS akun Anda. Anda tidak dapat membagikan Reservasi Kapasitas yang telah dibagikan dengan Anda.
- Anda hanya dapat berbagi Reservasi Kapasitas untuk instans penghunian bersama. Anda tidak dapat membagikan Reservasi Kapasitas untuk instans penghunian khusus.
- Kapasitas Berbagi reservasi tidak tersedia untuk AWS akun baru atau AWS akun yang memiliki riwayat penagihan terbatas.
- Untuk berbagi Reservasi Kapasitas dengan AWS organisasi atau unit organisasi di AWS organisasi Anda, Anda harus mengaktifkan berbagi dengan AWS Organizations. Untuk informasi selengkapnya, lihat Aktifkan Berbagi dengan AWS Organizations dalam Panduan Pengguna AWS RAM .
- Anda dapat berbagi Reservasi Kapasitas di `active` atau `scheduled` negara bagian. Anda tidak dapat berbagi Reservasi Kapasitas di negara bagian lain, seperti `assessing` atau`unsupported`.

Layanan terkait

Kapasitas Pembagian reservasi terintegrasi dengan AWS Resource Access Manager (AWS RAM). AWS RAM adalah layanan yang memungkinkan Anda untuk berbagi AWS sumber daya Anda dengan AWS akun apa pun atau melalui AWS Organizations. Dengan AWS RAM, Anda dapat berbagi sumber daya yang Anda miliki dengan membuat berbagi sumber daya. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan dibagikan. Konsumen dapat berupa AWS akun individu, atau unit organisasi atau seluruh organisasi dari AWS Organizations.

Untuk informasi selengkap AWS RAM nya, lihat Panduan AWS RAM Pengguna .

Berbagi di seluruh Zona Ketersediaan

Untuk memastikan bahwa sumber daya didistribusikan di seluruh Zona Ketersediaan untuk suatu Wilayah, kami secara independen memetakan Zona Ketersediaan ke nama untuk setiap akun . Hal ini dapat menyebabkan perbedaan penamaan Zona Ketersediaan di seluruh akun. Misalnya, Zona Ketersediaan `us-east-1a` untuk AWS akun Anda mungkin tidak memiliki lokasi yang sama `us-east-1a` dengan AWS akun lain.

Untuk mengidentifikasi lokasi Reservasi Kapasitas Anda yang terkait dengan akun Anda, Anda harus menggunakan ID Zona Ketersediaan (AZ ID). ID AZ adalah pengenal unik dan konsisten untuk Zona Ketersediaan di semua AWS akun. Misalnya, `use1-az1` adalah ID AZ untuk Wil `us-east-1` ayah dan itu adalah lokasi yang sama di setiap AWS akun.

Saat Anda berbagi Reservasi Kapasitas dengan akun lain, Reservasi Kapasitas dikaitkan dengan lokasi tertentu yang diidentifikasi oleh ID AZ-nya. Akun konsumen dapat menggunakan Reservasi Kapasitas bersama hanya di Zona Ketersediaan yang dipetakan ke ID AZ yang sama di akun mereka. Misalnya, jika Anda membuat Reservasi Kapasitas di `us-east-1a` (ID AZ`use1-az1`), konsumen harus meluncurkan instans di Zona Ketersediaan yang dipetakan ke `use1-az1` akun mereka. Zona Ketersediaan itu mungkin memiliki nama yang berbeda, seperti`us-east-1b`.

Untuk melihat ID AZ untuk Zona Ketersediaan di akun Anda

1. Buka AWS RAM konsol di https://console.aws.amazon.com/ram/home.
2. ID AZ untuk Wilayah saat ini ditampilkan di panel ID AZ Anda di sisi kanan layar.

### Berbagi Reservasi Kapasitas

Saat berbagi Reservasi Kapasitas yang Anda miliki dengan orang lain Akun AWS, Anda mengaktifkan mereka untuk meluncurkan instans ke kapasitas cadangan Anda. Jika Anda berbagi Reservasi Kapasitas terbuka, perhatikan hal berikut karena dapat mengakibatkan penggunaan Reservasi Kapasitas yang tidak diinginkan:

- Jika konsumen memiliki instans berjalan yang cocok dengan atribut Reservasi Kapasitas, mengatur parameter `CapacityReservationPreference` ke `open`, tetapi belum berjalan dalam kapasitas terpesan, mereka secara otomatis menggunakan Reservasi Kapasitas bersama.
- Jika konsumen meluncurkan instans yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa) dan `CapacityReservationPreference` parameter disetel ke`open`, mereka akan secara otomatis meluncurkan ke Reservasi Kapasitas bersama.

Untuk membagikan Reservasi Kapasitas, Anda harus menambahkannya ke berbagi sumber daya. Berbagi sumber daya adalah AWS RAM sumber daya yang memungkinkan Anda berbagi sumber daya di seluruh AWS akun. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan berbagi dengan mereka. Saat Anda berbagi Reservasi Kapasitas menggunakan konsol Amazon EC2, Anda menambahkannya ke berbagi sumber daya

yang ada. Untuk menambahkan Reservasi Kapasitas ke berbagi sumber daya baru, Anda harus membuat pembagian sumber daya menggunakan konsol AWS RAM.

Jika Anda adalah bagian dari organisasi AWS Organizations dan berbagi dalam organisasi Anda diaktifkan, konsumen di organisasi Anda diberikan akses ke Reservasi Kapasitas bersama jika prasyarat untuk berbagi terpenuhi. Jika Reservasi Kapasitas dibagikan dengan akun eksternal, mereka menerima undangan untuk bergabung dengan berbagi sumber daya dan diberikan akses ke Reservasi Kapasitas bersama setelah menerima undangan.

> **Important**
>
> Sebelum meluncurkan instans ke Reservasi Kapasitas yang dibagikan dengan Anda, verifikasi bahwa Anda memiliki akses ke Reservasi Kapasitas bersama dengan melihatnya di konsol atau dengan menjelaskannya menggunakan perintah describe-capacity-reservations. Jika Anda dapat melihat Reservasi Kapasitas bersama di konsol atau menjelaskannya menggunakan AWS CLI, itu tersedia untuk Anda gunakan dan Anda dapat meluncurkan instans ke dalamnya. Jika Anda mencoba meluncurkan instans ke Reservasi Kapasitas dan instans tersebut tidak dapat diakses karena kegagalan berbagi, instans akan diluncurkan ke On-Demand kapasitas.

Console

Untuk membagikan Reservasi Kapasitas yang Anda miliki menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Reservasi Kapasitas untuk dibagikan dan pilih Tindakan, Bagikan reservasi.
4. Pilih bagian sumber daya yang ingin ditambahkan Reservasi Kapasitas dan pilih Bagikan Reservasi Kapasitas.

   Butuh beberapa menit bagi konsumen untuk mendapatkan akses ke Reservasi Kapasitas bersama.

Untuk berbagi Reservasi Kapasitas yang Anda miliki menggunakan AWS RAM konsol

Lihat Membuat berbagi sumber daya di Panduan AWS RAM Pengguna.

AWS CLI

Untuk berbagi Reservasi Kapasitas yang Anda miliki

Gunakan perintah create-resource-share.

```
aws ram create-resource-share \
    --name my-resource-share \
    --resource-arns arn:aws:ec2:us-east-2:123456789012:capacity-
reservation/cr-1234abcd56EXAMPLE
```

PowerShell

Untuk berbagi Reservasi Kapasitas yang Anda miliki

Gunakan New-RAMResourceShare cmdlet.

```
New-RAMResourceShare `
    -Name my-resource-share `
    -ResourceArn "arn:aws:ec2:us-east-2:123456789012:capacity-
reservation/cr-1234abcd56EXAMPLE"
```

Berhenti membagikan Reservasi Kapasitas

Pemilik Reservasi Kapasitas dapat berhenti membagikan Reservasi Kapasitas kapan saja. Aturan-aturan berikut berlaku:

- Instans yang dimiliki oleh konsumen yang berjalan dalam kapasitas bersama pada saat berbagi berhenti terus berjalan dalam kapasitas yang dicadangkan. Kapasitas dikembalikan ke Reservasi Kapasitas saat konsumen menghentikan instans.
- Konsumen dengan siapa Reservasi Kapasitas dibagikan tidak dapat lagi meluncurkan instans baru ke dalam kapasitas terpesan.

Untuk berhenti berbagi Reservasi Kapasitas yang Anda miliki, Anda harus menghapusnya dari berbagi sumber daya.

Console

Untuk membagikan Reservasi Kapasitas yang Anda miliki menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Reservasi Kapasitas dan pilih tab Berbagi.
4. Tab Berbagi menampilkan daftar sumber daya tempat Reservasi Kapasitas ditambahkan. Pilih bagian sumber daya tempat Reservasi Kapasitas dihapus dan pilih Hapus dari pembagian sumber daya.

Untuk berhenti berbagi Reservasi Kapasitas yang Anda miliki menggunakan AWS RAM konsol

Lihat Memperbarui Sumber Daya Bersama di Panduan Pengguna AWS RAM .

AWS CLI

Untuk menghentikan berbagi Reservasi Kapasitas yang Anda miliki

Gunakan perintah disassociate-resource-share.

```
aws ram disassociate-resource-share \
    --resource-share-arn arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE \
 --resource-arns arn:aws:ec2:us-east-2:123456789012:capacity-reservation/cr-1234abcd56EXAMPLE
```

PowerShell

Untuk menghentikan berbagi Reservasi Kapasitas yang Anda miliki

Gunakan Disconnect-RAMResourceShare cmdlet.

```
Disconnect-RAMResourceShare `
    -ResourceShareArn "arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE" `
    -ResourceArn "arn:aws:ec2:us-east-2:123456789012:capacity-reservation/cr-1234abcd56EXAMPLE"
```

## Penetapan penagihan untuk Reservasi Kapasitas Amazon EC2 bersama

Secara default, ketika Reservasi Kapasitas dibagikan, pemilik akan ditagih untuk instans yang mereka jalankan di Reservasi Kapasitas dan untuk kapasitas apa pun yang tersedia, juga disebut kapasitas yang tidak digunakan, di Reservasi Kapasitas; sementara konsumen hanya ditagih untuk instans yang mereka jalankan di Reservasi Kapasitas bersama.

Jika diperlukan, pemilik Reservasi Kapasitas dapat menetapkan penagihan kapasitas apa pun yang tersedia di Reservasi Kapasitas ke salah satu akun yang digunakan untuk berbagi Reservasi Kapasitas. Setelah penagihan ditetapkan ke akun lain, akun tersebut menjadi pemilik penagihan dari kapasitas apa pun yang tersedia di Reservasi Kapasitas. Setiap biaya untuk kapasitas yang tersedia di Reservasi Kapasitas, sejak saat itu dan seterusnya, ditagih ke akun yang ditetapkan, bukan akun pemilik. Pemilik Reservasi Kapasitas dan akun yang digunakan bersama Reservasi Kapasitas terus ditagih untuk instans yang dijalankan di Reservasi Kapasitas.

> **Important**
>
> Pemilik Reservasi Kapasitas tetap menjadi pemilik sumber daya dan mereka tetap bertanggung jawab untuk mengelola Reservasi Kapasitas. Akun yang ditugaskan untuk penagihan tidak mendapatkan hak istimewa tambahan; mereka tidak dapat membatalkan, mengubah, atau membagikan Reservasi Kapasitas dengan cara apa pun.

Topik


- Cara kerjanya
- Pertimbangan-pertimbangan
- Menetapkan penagihan Reservasi Kapasitas EC2 bersama ke akun lain
- Melihat permintaan penugasan penagihan untuk Reservasi Kapasitas EC2 bersama
- Menerima atau menolak penagihan Reservasi Kapasitas EC2 bersama
- Membatalkan atau mencabut permintaan penugasan penagihan untuk Reservasi Kapasitas EC2 bersama
- Memantau permintaan penugasan penagihan untuk Reservasi Kapasitas bersama


### Cara kerjanya

Hanya pemilik Reservasi Kapasitas yang dapat menetapkan penagihan Reservasi Kapasitas bersama ke akun lain. Penagihan hanya dapat ditetapkan ke akun yang dibagikan Reservasi

Kapasitas dan yang dikonsolidasikan di bawah akun pem AWS Organizations bayar yang sama dengan pemilik Reservasi Kapasitas.

Untuk menetapkan penagihan kapasitas yang tersedia dari Reservasi Kapasitas ke akun lain, pemilik Reservasi Kapasitas harus memulai permintaan ke akun yang diperlukan. Akun yang ditentukan menerima permintaan dan mereka harus menerima atau menolaknya dalam waktu 12 jam.

- Jika mereka menerima, mereka menjadi pemilik penagihan dari kapasitas apa pun yang tersedia, juga disebut kapasitas yang tidak digunakan, di Reservasi Kapasitas. Sejak saat itu dan seterusnya, biaya untuk kapasitas apa pun yang tersedia di Reservasi Kapasitas ditagih ke akun mereka, bukan akun pemilik. Setelah diterima, hanya pemilik Reservasi Kapasitas yang dapat mencabut penagihan dari akun yang ditetapkan.
- Jika mereka menolak, pemilik Reservasi Kapasitas tetap menjadi pemilik penagihan kapasitas yang tersedia di Reservasi Kapasitas. Biaya untuk kapasitas apa pun yang tersedia di Reservasi Kapasitas terus ditagih ke akun pemilik.
- Jika mereka tidak menerima atau menolak permintaan dalam waktu 12 jam, permintaan akan kedaluwarsa dan biaya untuk kapasitas apa pun yang tersedia dalam Reservasi Kapasitas akan terus ditagih ke akun pemilik.

Untuk periode penagihan ditetapkan ke akun lain, item `Reservation` `UnusedBox` baris dan muncul di Laporan Biaya dan Penggunaan (CUR) akun yang ditetapkan, bukan CUR pemilik.

Tabel berikut menunjukkan item baris mana yang muncul di CUR untuk pemilik Reservasi Kapasitas dan akun konsumen sebelum penagihan ditetapkan ke akun lain.

| Akun | Item baris CUR sebelum penagihan ditetapkan |
| --- | --- |
| Kapasitas Pemilik Reservasi | • `Reservation`<br>• `BoxUsage *`<br>• `UnusedBox` |

| Akun | Item baris CUR sebelum penagihan ditetapkan |
| --- | --- |
| Akun konsumen yang digunakan untuk berbagi Reservasi Kapasitas | • `BoxUsage *` |

Tabel berikut menunjukkan item baris mana yang muncul di CUR untuk pemilik Reservasi Kapasitas dan akun konsumen setelah penagihan ditetapkan ke akun lain.

| Akun | Item baris CUR setelah penagihan ditetapkan |
| --- | --- |
| Kapasitas Pemilik Reservasi | • `BoxUsage *` |
| Akun konsumen yang ditugaskan penagihan | • `Reservation`<br>• `BoxUsage *`<br>• `UnusedBox` |
| Akun konsumen lain yang digunakan untuk berbagi Reservasi Kapasitas | • `BoxUsage *` |

**Note**

- * Item `BoxUsage` baris muncul di CUR akun hanya jika mereka memiliki instans yang sedang berjalan di Reservasi Kapasitas. Untuk informasi selengkapnya tentang item baris CUR, lihat Meman tau Reservasi Kapasitas.

- Gunakan ARN Reservasi Kapasitas di CUR untuk menentukan siapa yang memiliki Reservasi Kapasitas. Jika ARN menyertakan ID AWS akun Anda, Anda adalah pemilik Reservasi Kapasitas. Jika tidak, Reservasi Kapasitas dimiliki oleh akun yang berbeda tetapi penagihan ditetapkan kepada Anda.
- Tag alokasi biaya yang ditetapkan ke Reservasi Kapasitas oleh pemilik tidak akan muncul di CUR akun konsumen. Tag alokasi biaya hanya muncul di CUR pemilik Reservasi Kapasitas.

Pertimbangan-pertimbangan

Ingatlah hal-hal berikut saat menetapkan penagihan Reservasi Kapasitas bersama:

- Anda tidak dapat melakukan penugasan penagihan sebagian atau terpisah. Penagihan semua kapasitas yang tersedia dari Reservasi Kapasitas dapat ditetapkan ke satu akun pada satu waktu.
- Kapasitas yang tersedia dari Reservasi Kapasitas dapat berubah seiring waktu. Ini akan memengaruhi penagihan untuk akun yang ditetapkan. Misalnya, kapasitas yang tersedia dapat meningkat jika pemilik Reservasi Kapasitas meningkatkan ukuran Reservasi Kapasitas, atau jika akun konsumen lain menghentikan atau menghentikan instans mereka yang berjalan di Reservasi Kapasitas.
- Penagihan hanya dapat ditetapkan ke akun konsumen yang dikonsolidasikan di bawah akun pem AWS Organizations bayar yang sama. Penagihan secara otomatis dicabut dari akun konsumen jika mereka meninggalkan organisasi, atau jika Reservasi Kapasitas tidak lagi dibagikan dengan mereka.
- Hanya pemilik Reservasi Kapasitas yang dapat membatalkan permintaan penugasan penagihan yang tertunda dan mencabut penagihan dari akun yang ditetapkan setelah permintaan diterima.
- Jika pemilik membatalkan Reservasi Kapasitas bertanggal mendatang yang memiliki penagihan yang ditetapkan ke akun lain, pemilik tagihan yang ditugaskan membayar biaya pembatalan.

Menetapkan penagihan Reservasi Kapasitas EC2 bersama ke akun lain

Untuk menetapkan penagihan kapasitas yang tersedia dari Reservasi Kapasitas bersama ke akun lain, pemilik Reservasi Kapasitas harus memulai permintaan ke akun yang diperlukan. Di konsol Amazon EC2, permintaan ini disebut permintaan transfer.

Pemilik Reservasi Kapasitas dapat menetapkan penagihan kapasitas Reservasi Kapasitas yang tersedia ke akun hanya jika:

- Reservasi Kapasitas sudah dibagikan dengan akun itu.
- Akun dikonsolidasikan di bawah akun pem AWS Organizations bayar yang sama dengan pemilik Reservasi Kapasitas.

Penagihan ditetapkan ke akun yang ditentukan hanya setelah mereka menerima permintaan.

Saat pemilik Reservasi Kapasitas memulai permintaan, EventBridge acara Amazon dikirim ke akun yang diminta. Untuk informasi selengkapnya, lihat Memantau permintaan penugasan penagihan untuk Reservasi Kapasitas bersama.

Console

Untuk menetapkan penagihan Reservasi Kapasitas bersama

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas dan kemudian pilih Reservasi Kapasitas bersama.
3. Di bagian Pen agihan kapasitas yang tersedia, pilih Tet apkan penagihan.
4. Di layar Tetapkan penagihan, pilih akun konsumen yang akan ditugaskan penagihan, lalu pilih Per mintaan.

AWS CLI

Untuk menetapkan penagihan Reservasi Kapasitas bersama

Gunakan perintah associate-capac ity-reservation-billing-owner. Untuk`--capacity-reservation-id`, tentukan ID Reservasi Kapasitas bersama. Untuk`--unused-reservation-billing-owner-id`, tentukan ID AWS akun yang akan ditetapkan penagihan.

```
aws ec2 associate-capacity-reservation-billing-owner \
    --capacity-reservation-id cr-01234567890abcdef \
    --unused-reservation-billing-owner-id 123456789012
```

PowerShell

Untuk menetapkan penagihan Reservasi Kapasitas bersama

Gunakan Register-EC2CapacityReservationBillingOwner cmdlet. Untuk`-CapacityReservationId`, tentukan ID Reservasi Kapasitas bersama. Untuk-

`UnusedReservationBillingOwnerId`, tentukan ID AWS akun yang akan ditetapkan penagihan.

```
Register-EC2CapacityReservationBillingOwner `
    -CapacityReservationId cr-01234567890abcdef `
    -UnusedReservationBillingOwnerId 123456789012
```

Melihat permintaan penugasan penagihan untuk Reservasi Kapasitas EC2 bersama

Pemilik Reservasi Kapasitas hanya dapat melihat permintaan penetapan penagihan terbaru yang mereka mulai. Dan akun konsumen hanya dapat melihat permintaan penugasan penagihan terbaru yang dikirim kepada mereka.

Permintaan dapat dilihat selama 24 jam setelah mereka memasuki`cancelled`,`expired`, atau `revoked` negara bagian. Setelah 24 jam, mereka tidak dapat lagi dilihat.

Console

(Pemilik Reservasi Kapasitas) Untuk melihat permintaan yang Anda mulai

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reserv asi Kapasitas, lalu pilih Reservasi Kapasitas bersama untuk melihat permintaan.
3. Bagian Penagihan kapasitas yang tersedia menunjukkan permintaan terbaru dan statusnya saat ini.

(Akun konsumen) Untuk permintaan yang dikirimkan kepada Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Jika Anda memiliki permintaan yang tertunda, spanduk Permintaan penugasan penagihan yang tertunda akan muncul di bagian atas layar. Jika spanduk tidak muncul, Anda tidak memiliki permintaan yang tertunda.

   Untuk melihat permintaan, pilih Permintaan Tinjau di spanduk.

AWS CLI

(Pemilik Reservasi Kapasitas) Untuk melihat permintaan yang Anda mulai

Gunakan perintah describe-capacity-reservation-billing-requests.

```
aws ec2 describe-capacity-reservation-billing-requests \
    --role odcr-owner
```

(Akun konsumen) Untuk melihat permintaan yang dikirimkan kepada Anda

Gunakan perintah describe-capacity-reservation-billing-requests.

```
aws ec2 describe-capacity-reservation-billing-requests \
    --role unused-reservation-billing-owner
```

PowerShell

(Pemilik Reservasi Kapasitas) Untuk melihat permintaan yang Anda mulai

Gunakan Get-EC2CapacityReservationBillingRequest cmdlet.

```
Get-EC2CapacityReservationBillingRequest `
    -Role odcr-owner
```

(Akun konsumen) Untuk melihat permintaan yang dikirimkan kepada Anda

Gunakan Get-EC2CapacityReservationBillingRequest cmdlet.

```
Get-EC2CapacityReservationBillingRequest `
    -Role unused-reservation-billing-owner
```

Permintaan dapat berada di salah satu negara bagian berikut.

| Status | Deskripsi | | | |
|---|---|---|---|---|
| `pending` | Permintaan belum diterima atau ditolak, tetapi belum kedaluwarsa. | | | |

| Status | Deskripsi |
| --- | --- |
| `accepted` | Permintaan diterima oleh akun yang ditentukan. Penagihan kapasitas yang tersedia dari Reservasi Kapasitas ditetapkan ke akun konsumen. |
| `rejected` | Permintaan ditolak oleh akun konsumen. |
| `cancelled` | Permintaan dibatalkan oleh pemilik Reservasi Kapasitas saat berada di `pending` negara bagian. |
| `revoked` | Penagihan dicabut dari akun konsumen karena salah satu alasan berikut:<br>• Itu secara eksplisit dicabut oleh pemilik Reservasi Kapasitas.<br>• Reservasi Kapasitas tidak lagi dibagikan dengan akun konsumen.<br>• Akun konsumen tidak lagi menjadi bagian dari AWS organisasi. |
| `expired` | Permintaan kedaluwarsa karena akun konsumen tidak menerima atau menolaknya dalam waktu 12 jam. |

### Menerima atau menolak penagihan Reservasi Kapasitas EC2 bersama

Jika Anda menerima permintaan penugasan penagihan untuk Reservasi Kapasitas yang dibagikan dengan Anda, Anda dapat menerima atau menolaknya. Permintaan tetap di `pending` negara bagian sampai diterima atau ditolak.

Jika Anda menerima permintaan tersebut, permintaan akan masuk ke `accepted` status, dan penagihan kapasitas apa pun yang tersedia, atau tidak digunakan, dari Reservasi Kapasitas tersebut ditetapkan ke akun Anda sejak saat itu dan seterusnya. Setelah Anda menerima permintaan, hanya pemilik Reservasi Kapasitas yang dapat mencabut penagihan dari akun Anda.

Jika Anda menolak permintaan, permintaan akan masuk ke `rejected` status, dan penagihan kapasitas yang tersedia dari Reservasi Kapasitas tetap ditetapkan kepada pemilik Reservasi Kapasitas.

Permintaan akan kedaluwarsa jika tidak diterima atau ditolak dalam waktu 12 jam. Jika permintaan kedaluwarsa, penagihan kapasitas yang tidak terpakai dari Reservasi Kapasitas tetap ditetapkan kepada pemilik Reservasi Kapasitas.

Ketika permintaan diterima atau ditolak, EventBridge acara Amazon dikirim ke akun pemilik Reservasi Kapasitas. Ketika permintaan kedaluwarsa, EventBridge acara Amazon dikirim ke pemilik Reservasi Kapasitas dan akun konsumen. Untuk informasi selengkapnya, lihat Memantau permintaan penugasan penagihan untuk Reservasi Kapasitas bersama.

Console

Menerima atau menolak permintaan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Jika Anda memiliki permintaan yang tertunda, spanduk Permintaan penugasan penagihan yang tertunda akan muncul di bagian atas layar. Jika spanduk tidak muncul, Anda tidak memiliki permintaan yang tertunda.

   Untuk melihat permintaan, pilih Permintaan Tinjau di spanduk.
4. Pilih permintaan untuk menerima atau ditolak, lalu pilih Terima atau Tol ak.

AWS CLI

Untuk menerima permintaan

Gunakan perintah accept-capacity-reservation-billing-ownership. Untuk `--capacity-reservation-id`, tentukan ID Reservasi Kapasitas yang akan menerima permintaan.

```
aws ec2 accept-capacity-reservation-billing-ownership \
    --capacity-reservation-id cr-01234567890abcdef
```

Untuk menolak permintaan

Gunakan perintah reject-capacity-reservation-billing-ownership. Untuk `--capacity-reservation-id`, tentukan ID Reservasi Kapasitas yang akan ditolak permintaan.

```
aws ec2 reject-capacity-reservation-billing-ownership \
    --capacity-reservation-id cr-01234567890abcdef
```

PowerShell

Untuk menerima permintaan

Gunakan Approve-EC2CapacityReservationBillingOwnership cmdlet.

```
Approve-EC2CapacityReservationBillingOwnership `
    -CapacityReservationId cr-01234567890abcdef
```

Untuk menolak permintaan

Gunakan Deny-EC2CapacityReservationBillingOwnership cmdlet.

```
Deny-EC2CapacityReservationBillingOwnership `
    -CapacityReservationId cr-01234567890abcdef
```

Membatalkan atau mencabut permintaan penugasan penagihan untuk Reservasi Kapasitas EC2 bersama

Hanya pemilik Reservasi Kapasitas yang dapat membatalkan permintaan `pending` penugasan penagihan. Jika permintaan yang tertunda dibatalkan, permintaan tersebut akan masuk ke `cancelled` status dan penagihan kapasitas reservasi kapasitas yang tersedia, atau tidak digunakan, tetap ditetapkan kepada pemilik Reservasi Kapasitas.

Setelah permintaan dibuat`accepted`, hanya pemilik Reservasi Kapasitas yang dapat mencabut penagihan dari akun yang ditetapkan. Jika penagihan dicabut, permintaan akan masuk ke `revoked`

status dan penagihan kapasitas yang tersedia dari Reservasi Kapasitas dialihkan kepada pemilik Reservasi Kapasitas.

Saat permintaan dibatalkan atau dicabut, EventBridge acara Amazon dikirim ke pemilik Reservasi Kapasitas dan akun konsumen tertentu. Untuk informasi selengkapnya, lihat Memantau permintaan penugasan penagihan untuk Reservasi Kapasitas bersama.

Console

Untuk membatalkan atau mencabut permintaan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas, lalu pilih Reservasi Kapasitas untuk membatalkan atau mencabut permintaan.
3. Di bagian Penagihan kapasitas yang tersedia, pilih Batalkan transfer atau Cabut transfer, tergantung pada status permintaan saat ini.

AWS CLI

Untuk membatalkan atau mencabut permintaan

Gunakan perintah disassociate-capacity-reservation-billing-owner. Untuk `--unused-reservation-billing-owner-id`, tentukan ID AWS akun tempat permintaan dikirim.

```
aws ec2 disassociate-capacity-reservation-billing-owner \
    --capacity-reservation-id cr-01234567890abcdef \
    --unused-reservation-billing-owner-id 123456789012
```

PowerShell

Untuk membatalkan atau mencabut permintaan

Gunakan Unregister-EC2CapacityReservationBillingOwner cmdlet. Untuk `-UnusedReservationBillingOwnerId`, tentukan ID AWS akun tempat permintaan dikirim.

```
Unregister-EC2CapacityReservationBillingOwner `
    -CapacityReservationId cr-01234567890abcdef `
    -UnusedReservationBillingOwnerId 123456789012
```

Memantau permintaan penugasan penagihan untuk Reservasi Kapasitas bersama

Amazon EC2 mengirimkan EventBridge peristiwa Amazon saat status permintaan penugasan penagihan berubah.

- Acara dikirim ke pemilik Reservasi Kapasitas ketika permintaan memasuki status berikut: `accepted` | `rejected` | `expired` |`revoked`.
- Acara dikirim ke akun konsumen yang diminta ketika permintaan memasuki status berikut: `pending` | `expired` | `cancelled` |`revoked`.

Untuk informasi selengkapnya tentang Amazon EventBridge, lihat Panduan EventBridge Pengguna Amazon.

Berikut ini adalah pola EventBridge acara Amazon.

```
{
   "version":"0",
   "id":"12345678-1234-1234-1234-123456789012",
   "detail-type":"On-Demand Capacity Reservation Billing Ownership Request pending|
accepted|rejected|cancelled|revoked|expired",
   "source":"aws.ec2",
   "account":"account_id",
   "time":"state_change_timestamp",
   "region":"region",
   "resources":[
      "arn:aws:ec2:region:cr_owner_account_id:capacity-reservation/cr_id"
   ],
   "detail":{
      "capacity-reservation-id":"cr_id",
      "updateTime":timestamp,
      "ownerAccountId":"cr_owner_account_id",
      "unusedReservationChargesOwnerID":"consumer_account_id",
      "status":"pending|accepted|rejected|cancelled|revoked|expired",
      "statusMessage":"message
   }
}
```

Berikut ini adalah contoh peristiwa yang dikirim ke pemilik Reservasi Kapasitas (`222222222222`) ketika akun konsumen (`111111111111`) menerima permintaan penetapan penagihan untuk Reservasi Kapasitas bersama (`cr-01234567890abcdef`).

```
{
   "version":"0",
   "id":"12345678-1234-1234-1234-123456789012",
   "detail-type":"On-Demand Capacity Reservation Billing Ownership Request accepted",
   "source":"aws.ec2",
   "account":"222222222222",
   "time":"2024-09-01Thh:59:59Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ec2:us-east-1:222222222222:capacity-reservation/cr-01234567890abcdef"
   ],
   "detail":{
      "capacity-reservation-id":"cr-01234567890abcdef",
      "updateTime":"2024-08-01Thh:59:59Z",
      "ownerAccountId":"222222222222",
      "unusedReservationChargesOwnerID":"111111111111",
      "status":"accepted",
      "statusMessage":"billing transfer status message"
   }
}
```

Izin Reservasi Kapasitas Bersama

Izin untuk pemilik

Pemilik bertanggung jawab untuk mengelola dan membatalkan Reservasi Kapasitas bersama mereka. Pemilik tidak dapat memodifikasi instans yang berjalan di Reservasi Kapasitas bersama yang dimiliki oleh akun lain. Pemilik tetap bertanggung jawab untuk mengelola instans yang mereka luncurkan ke dalam Reservasi Kapasitas bersama.

Izin untuk konsumen

Konsumen bertanggung jawab untuk mengelola instans mereka yang menjalankan Reservasi Kapasitas bersama. Konsumen tidak dapat memodifikasi Reservasi Kapasitas bersama dengan cara apa pun, dan mereka tidak dapat menampilkan atau memodifikasi instans yang dimiliki oleh konsumen lain atau pemilik Reservasi Kapasitas. Konsumen hanya dapat melihat total kapasitas dan kapasitas yang tersedia dalam reservasi bersama.

Tagihan dan pengukuran

Tidak ada biaya tambahan untuk berbagi Reservasi Kapasitas.

Secara default, pemilik Reservasi Kapasitas ditagih untuk instans yang dijalankan di dalam Reservasi Kapasitas dan untuk kapasitas cadangan yang tidak digunakan, sementara konsumen ditagih untuk instans yang mereka jalankan di dalam Reservasi Kapasitas bersama. Namun, Anda dapat menetapkan penagihan kapasitas yang tersedia dari Reservasi Kapasitas bersama ke akun konsumen tertentu. Untuk informasi selengkapnya, lihat Penetapan penagihan untuk Reservasi Kapasitas Amazon EC2 bersama.

Jika pemilik Reservasi Kapasitas termasuk dalam rekening pembayar yang berbeda dan Reservasi Kapasitas tercakup oleh Instans Terpesan Regional atau Savings Plans, pemilik Reservasi Kapasitas akan terus ditagih untuk Instans Terpesan Regional atau Savings Plans. Dalam kasus ini, pemilik Reservasi Kapasitas membayar Instans Terpesan Regional atau Savings Plans, dan konsumen ditagih untuk instans yang dijalankan dalam Reservasi Kapasitas bersama.

Batas instans

Semua penggunaan Reservasi Kapasitas diperhitungkan dalam batas On-Demand Instans pemilik Reservasi Kapasitas. Hal ini mencakup:

- Kapasitas terpesan yang tidak terpakai
- Penggunaan oleh instans yang dimiliki oleh pemilik Reservasi Kapasitas
- Penggunaan oleh instans yang dimiliki oleh konsumen

Instans yang diluncurkan ke kapasitas bersama oleh konsumen diperhitungkan dalam batas On-Demand Instans pemilik Reservasi Kapasitas. Batas instans konsumen adalah jumlah batas Inst On-Demand ans mereka sendiri dan kapasitas yang tersedia di Reservasi Kapasitas bersama yang dapat mereka akses.

Armada Reservasi Kapasitas

Armada Reservasi  On-Demand Kapasitas  adalah kelompok Reservasi Kapasitas.

Permintaan Armada Reservasi Kapasitas berisi semua informasi konfigurasi yang diperlukan untuk meluncurkan Armada Reservasi Kapasitas. Menggunakan satu permintaan, Anda dapat memesan sejumlah besar kapasitas Amazon EC2 untuk beban kerja di banyak tipe instans, hingga kapasitas target yang Anda tentukan.

Setelah membuat Armada Reservasi Kapasitas, Anda dapat mengelola Reservasi Kapasitas dalam armada secara kolektif dengan memodifikasi atau membatalkan Armada Reservasi Kapasitas.

Topik


- Cara kerja Armada Reservasi Kapasitas
- Pertimbangan-pertimbangan
- Penetapan harga
- Konsep dan perencanaan Armada Reservasi Kapasitas
- Untuk mengubah Armada Reservasi Kapasitas
- Memodifikasi Armada Reservasi Kapasitas
- Membatalkan Armada Reservasi Kapasitas
- Contoh konfigurasi Armada Reservasi Kapasitas
- Menggunakan peran tertaut layanan untuk Armada Reservasi Kapasitas


## Cara kerja Armada Reservasi Kapasitas

Saat Anda membuat Armada Reservasi Kapasitas, Armada mencoba membuat Reservasi Kapasitas individual untuk memenuhi total kapasitas target yang Anda tentukan dalam permintaan Armada.

Jumlah instans yang diandalkan kapasitas terpesan Armada bergantung pada total kapasitas target dan bobot tipe instans yang Anda tentukan. Tipe instans untuk reservasi kapasitas tergantung pada strategi alokasi dan prioritas tipe instans yang Anda gunakan.

Jika ada kapasitas yang tidak mencukupi pada saat Armada dibuat, dan tidak dapat segera memenuhi kapasitas target totalnya, Armada secara asinkron mencoba untuk membuat Reservasi Kapasitas sampai berhasil memesan sejumlah kapasitas yang diminta.

Ketika Armada mencapai kapasitas target totalnya, Armada berusaha mempertahankan kapasitas itu. Jika Reservasi Kapasitas di Armada dibatalkan, Armada secara otomatis membuat satu atau lebih Reservasi Kapasitas, tergantung pada konfigurasi Armada Anda, untuk mengganti kapasitas yang hilang dan mempertahankan total kapasitas targetnya.

Reservasi Kapasitas di Armada tidak dapat dikelola secara individual. Reservasi tersebut harus dikelola secara kolektif dengan memodifikasi Armada. Saat Anda memodifikasi Armada, Reservasi Kapasitas di Armada diperbarui secara otomatis untuk mencerminkan perubahan tersebut.

Saat ini, Armada Reservasi Kapasitas mendukung kriteria pencocokan instans `open`, dan semua Reservasi Kapasitas yang diluncurkan oleh Armada secara otomatis menggunakan kriteria

pencocokan instans ini. Dengan kriteria ini, instans baru dan instans yang sudah ada yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa) secara otomatis berjalan di Reservasi Kapasitas yang dibuat oleh Armada. Armada Reservasi Kapasitas tidak mendukung kriteria pencocokan instans yang ditargetkan.

### Pertimbangan-pertimbangan

Ingatlah hal-hal berikut ini saat bekerja dengan Armada Reservasi Kapasitas:

- Armada Reservasi Kapasitas dapat dibuat, dimodifikasi, dilihat, dan dibatalkan menggunakan AWS CLI dan AWS API.
- Reservasi Kapasitas dalam suatu Armada tidak dapat dikelola secara individual. Reservasi tersebut harus dikelola secara kolektif dengan memodifikasi atau membatalkan Armada.
- Armada Reservasi Kapasitas tidak dapat menjangkau seluruh Wilayah.
- Armada Reservasi Kapasitas tidak dapat menjangkau seluruh Zona Ketersediaan.
- Reservasi Kapasitas yang dibuat oleh Armada Reservasi Kapasitas secara otomatis ditandai dengan tanda yang dihasilkan AWS berikut:
  - Kunci — `aws:ec2-capacity-reservation-fleet`
  - Nilai — *`fleet_id`*

  Anda dapat menggunakan tanda ini untuk mengidentifikasi Reservasi Kapasitas yang dibuat oleh Armada Reservasi Kapasitas.

### Penetapan harga

Tidak ada biaya tambahan untuk menggunakan Armada Reservasi Kapasitas. Anda ditagih untuk Reservasi Kapasitas individual yang dibuat oleh Armada Reservasi Kapasitas Anda. Untuk informasi selengkapnya tentang cara penagihan Reservasi Kapasitas, lihat Harga dan penagihan Reservasi Kapasitas.

## Konsep dan perencanaan Armada Reservasi Kapasitas

Informasi berikut menjelaskan cara merencanakan Armada Reservasi Kapasitas dan menjelaskan konsep Armada Reservasi Kapasitas termasuk kapasitas target total, strategi alokasi, bobot jenis instans, dan prioritas tipe instans.

### Topik

- Rencanakan Armada Reservasi Kapasitas
- Kapasitas target total
- Strategi alokasi
- Bobot tipe instans
- Prioritas tipe instans


## Rencanakan Armada Reservasi Kapasitas

Saat merencanakan Armada Reservasi Kapasitas Anda, kami sarankan Anda melakukan hal berikut:

1. Tentukan jumlah kapasitas komputasi yang dibutuhkan oleh beban kerja Anda.
2. Tentukan tipe instans dan Zona Ketersediaan yang ingin Anda gunakan.
3. Tetapkan prioritas untuk setiap tipe instans berdasarkan kebutuhan dan preferensi Anda. Untuk informasi selengkapnya, lihat Prioritas tipe instans.
4. Buat sistem pembobotan kapasitas yang masuk akal untuk beban kerja Anda. Tetapkan bobot untuk setiap tipe instans dan tentukan total kapasitas target Anda. Untuk informasi selengkapnya, lihat Bobot tipe instans dan Kapasitas target total.
5. Tentukan apakah Anda memerlukan Reservasi Kapasitas tanpa batas waktu atau hanya untuk jangka waktu tertentu.

## Kapasitas target total

Total kapasitas target menentukan jumlah total kapasitas komputasi yang dipesan oleh Armada Reservasi Kapasitas. Anda menentukan total kapasitas target saat Anda membuat Armada Reservasi Kapasitas. Setelah Armada dibuat, Amazon EC2 secara otomatis membuat Reservasi Kapasitas untuk memesan kapasitas hingga total kapasitas target.

Jumlah instans kapasitas yang dipesan Armada Reservasi Kapasitas ditentukan oleh total kapasitas target dan bobot tipe instans yang Anda tentukan untuk setiap tipe instans di Armada Reservasi Kapasitas (`total target capacity/instance type weight=number of instances`).

Anda dapat menetapkan total kapasitas target berdasarkan unit yang berarti bagi beban kerja Anda. Misalnya, jika beban kerja Anda memerlukan sejumlah vCPU, Anda dapat menetapkan total kapasitas target berdasarkan jumlah vCPU yang diperlukan. Jika beban kerja Anda memerlukan `2048` vCPU, tentukan total kapasitas target `2048`, lalu tetapkan bobot tipe instans berdasarkan jumlah vCPU yang disediakan oleh tipe instans di Armada. Sebagai contoh, lihat Bobot tipe instans.

### Strategi alokasi

Strategi alokasi untuk Armada Reservasi Kapasitas Anda menentukan caranya memenuhi permintaan Anda untuk kapasitas terpesan dari spesifikasi tipe instans dalam konfigurasi Armada Capacity Reservation.

Saat ini, hanya strategi alokasi `prioritized` yang didukung. Dengan strategi ini, Armada Reservasi Kapasitas membuat Reservasi Kapasitas menggunakan prioritas yang telah Anda tetapkan untuk setiap spesifikasi tipe instans dalam konfigurasi Armada Reservasi Kapasitas. Nilai prioritas yang lebih rendah menunjukkan prioritas penggunaan yang lebih tinggi. Misalnya, Anda membuat Armada Reservasi Kapasitas yang menggunakan tipe instans dan prioritas berikut:

- `m4.16xlarge` — prioritas = 1
- `m5.16xlarge` — prioritas = 3
- `m5.24xlarge` — prioritas = 2

Armada pertama kali mencoba untuk membuat Reservasi Kapasitas untuk `m4.16xlarge`. Jika Amazon EC2 memiliki kapasitas `m4.16xlarge` yang tidak mencukupi, Armada berupaya membuat Reservasi Kapasitas untuk `m5.24xlarge`. Jika Amazon EC2 memiliki kapasitas `m5.24xlarge` yang tidak mencukupi, Armada membuat Reservasi Kapasitas untuk `m5.16xlarge`.

### Bobot tipe instans

Bobot tipe instans adalah bobot yang Anda tetapkan untuk setiap tipe instans di Armada Reservasi Kapasitas. Bobot menentukan berapa banyak unit kapasitas setiap instans dari tipe instans tertentu yang diperhitungkan dalam total kapasitas target Armada.

Anda dapat menetapkan bobot berdasarkan unit yang berarti bagi beban kerja Anda. Misalnya, jika beban kerja Anda memerlukan sejumlah vCPU tertentu, Anda dapat menetapkan bobot berdasarkan jumlah vCPU yang disediakan oleh setiap tipe instans di Armada Reservasi Kapasitas. Dalam hal ini, jika Anda membuat Armada Reservasi Kapasitas menggunakan instans `m4.16xlarge` dan `m5.24xlarge`, Anda akan menetapkan bobot yang sesuai dengan jumlah vCPU untuk setiap instans sebagai berikut:

- `m4.16xlarge` — 64 vCPU, bobot = 64 unit
- `m5.24xlarge` — 96 vCPU, bobot = 96 unit

Bobot tipe instans menentukan jumlah instans yang kapasitasnya dipesan Armada Reservasi Kapasitas. Misalnya, jika Armada Reservasi Kapasitas dengan total kapasitas target 384 unit menggunakan jenis dan bobot instans dalam contoh sebelumnya, Armada dapat mencadangkan kapasitas untuk 6 `m4.16xlarge` instans (384 total jenis inst capacity/64 ans target bobot = 6 instans), atau 4 `m5.24xlarge` instans (384/96 = 4).

Jika Anda tidak menetapkan bobot tipe instans, atau jika Anda menetapkan bobot tipe instans 1, total kapasitas target hanya didasarkan pada jumlah instans. Misalnya, jika Armada Reservasi Kapasitas dengan kapasitas target total 384 unit menggunakan tipe instans dalam contoh sebelumnya, tetapi menghilangkan bobot atau menentukan bobot 1 untuk kedua tipe instans, Armada dapat memesan kapasitas untuk 384 `m4.16xlarge` instans atau 384 `m5.24xlarge` instans.

Prioritas tipe instans

Prioritas tipe instans adalah nilai yang Anda tetapkan ke tipe instans di Armada. Prioritas digunakan untuk menentukan tipe instans mana yang ditentukan untuk Armada yang harus diprioritaskan untuk digunakan.

Nilai prioritas yang lebih rendah menunjukkan prioritas penggunaan yang lebih tinggi.

Untuk mengubah Armada Reservasi Kapasitas

Saat Anda membuat Armada Reservasi Kapasitas, Armada akan secara otomatis membuat Reservasi Kapasitas untuk tipe instans yang ditentukan dalam permintaan Armada, hingga total kapasitas target yang ditentukan. Jumlah instans kapasitas yang dipesan Armada Reservasi Kapasitas tergantung pada total kapasitas target dan bobot tipe instans yang Anda tentukan dalam permintaan. Untuk informasi selengkapnya, lihat Bobot tipe instans dan Kapasitas target total.

Saat membuat Armada, Anda harus menentukan tipe instans yang akan digunakan dan prioritas untuk masing-masing tipe instans tersebut. Untuk informasi selengkapnya, lihat Strategi alokasi dan Prioritas tipe instans.

> **Note**
>
> Per AWSServiceRoleForEC2CapacityReservationFleet an terkait layanan secara otomatis dibuat di akun Anda saat pertama kali Anda membuat Armada Reservasi Kapasitas. Untuk informasi selengkapnya, lihat Menggunakan peran tertaut layanan untuk Armada Reservasi Kapasitas.

Saat ini, Armada Reservasi Kapasitas hanya mendukung kriteria pencocokan instans `open`.

AWS CLI

Untuk mengubah Armada Reservasi Kapasitas

Gunakan perintah create-capac ity-reservation-fleet.

```
aws ec2 create-capacity-reservation-fleet \
    --total-target-capacity 24 \
    --allocation-strategy prioritized \
    --instance-match-criteria open \
    --tenancy default \
    --end-date 2021-12-31T23:59:59.000Z \
    --instance-type-specifications file://instanceTypeSpecification.json
```

Berikut ini adalah isi dari`instanceTypeSpecification.json`.

```
[
  {
    "InstanceType": "m5.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "Weight": 3.0,
    "AvailabilityZone":"us-east-1a",
    "EbsOptimized": true,
    "Priority" : 1
  }
]
```

Berikut ini adalah output contoh.

```
{
    "Status": "submitted",
    "TotalFulfilledCapacity": 0.0,
    "CapacityReservationFleetId": "crf-abcdef01234567890",
    "TotalTargetCapacity": 24
}
```

PowerShell

Untuk mengubah Armada Reservasi Kapasitas

Gunakan New-EC2CapacityReservationFleet cmdlet.

```
New-EC2CapacityReservationFleet `
    -TotalTargetCapacity 24 `
    -AllocationStrategy "prioritized" `
    -InstanceMatchCriterion "open" `
    -Tenancy "default" `
    -EndDate 2021-12-31T23:59:59.000Z `
    -InstanceTypeSpecification $specification
```

Spesifikasi didefinisikan sebagai berikut.

```
$specification = New-Object Amazon.EC2.Model.ReservationFleetInstanceSpecification
$specification.InstanceType = "m5.xlarge"
$specification.InstancePlatform = "Linux/UNIX"
$specification.Weight = 3.0
$specification.AvailabilityZone = "us-east-1a"
$specification.EbsOptimized = $true
$specification.Priority = 1
```

### Memodifikasi Armada Reservasi Kapasitas

Anda dapat memodifikasi total kapasitas target dan tanggal Armada Reservasi Kapasitas kapan saja. Saat Anda memodifikasi total kapasitas target Armada Reservasi Kapasitas, Armada secara otomatis membuat Reservasi Kapasitas baru, atau memodifikasi atau membatalkan Reservasi Kapasitas yang ada di Armada untuk memenuhi total kapasitas target yang baru. Ketika Anda memodifikasi tanggal akhir Armada, tanggal akhir untuk semua Reservasi Kapasitas individu akan diperbarui sesuai dengan modifikasi itu.

#### Pertimbangan-pertimbangan

- Setelah Anda memodifikasi Armada, statusnya beralih ke `modifying`. Anda tidak dapat mencoba modifikasi tambahan pada Armada saat berada dalam status `modifying`.
- Anda tidak dapat mengubah penghunian, Zona Ketersediaan, tipe instans, platform instans, prioritas, atau bobot yang digunakan oleh Armada Reservasi Kapasitas. Jika Anda perlu mengubah salah satu parameter ini, Anda mungkin perlu membatalkan Armada yang ada dan membuat armada baru dengan parameter yang diperlukan.
- Anda tidak dapat menentukan `--end-date` dan `--remove-end-date` dalam perintah yang sama.

AWS CLI

Untuk mengubah Armada Reservasi Kapasitas

Gunakan perintah modify- capacity-reservation-fleet.

Contoh 1: Ubah kapasitas target total

```
aws ec2 modify-capacity-reservation-fleet \
    --capacity-reservation-fleet-id crf-01234567890abcedf \
    --total-target-capacity 160
```

Contoh 2: Ubah tanggal akhir

```
aws ec2 modify-capacity-reservation-fleet \
    --capacity-reservation-fleet-id crf-01234567890abcedf \
    --end-date 2021-07-04T23:59:59.000Z
```

Contoh 3: Hapus tanggal akhir

```
aws ec2 modify-capacity-reservation-fleet \
    --capacity-reservation-fleet-id crf-01234567890abcedf \
    --remove-end-date
```

PowerShell

Untuk mengubah Armada Reservasi Kapasitas

Gunakan Edit-EC2CapacityReservationFleet cmdlet.

Contoh 1: Ubah kapasitas target total

```
Edit-EC2CapacityReservationFleet `
    -CapacityReservationFleetId crf-01234567890abcedf `
    -TotalTargetCapacity 160
```

Contoh 2: Ubah tanggal akhir

```
Edit-EC2CapacityReservationFleet `
    -CapacityReservationFleetId crf-01234567890abcedf `
    -EndDate 2021-07-04T23:59:59.000Z
```

Contoh 3: Hapus tanggal akhir

```
Edit-EC2CapacityReservationFleet `
    -CapacityReservationFleetId crf-01234567890abcedf `
    -RemoveEndDate
```

### Membatalkan Armada Reservasi Kapasitas

Bila Anda tidak lagi membutuhkan Armada Reservasi Kapasitas dan kapasitas dipesan, Anda dapat membatalkannya. Saat Anda membatalkan Armada, statusnya berubah menjadi `cancelled` dan Armada tidak dapat lagi membuat Reservasi Kapasitas baru. Selain itu, semua Reservasi Kapasitas individu di Armada dibatalkan. Instans yang sebelumnya berjalan dalam kapasitas cadangan terus berjalan normal dalam kapasitas bersama.

AWS CLI

Untuk membatalkan Armada Reservasi Kapasitas

Gunakan perintah cancel-capac ity-reservation-fleets.

```
aws ec2 cancel-capacity-reservation-fleets \
    --capacity-reservation-fleet-ids crf-abcdef01234567890
```

Berikut ini adalah output contoh.

```
{
    "SuccessfulFleetCancellations": [
        {
            "CurrentFleetState": "cancelling",
            "PreviousFleetState": "active",
            "CapacityReservationFleetId": "crf-abcdef01234567890"
        }
    ],
    "FailedFleetCancellations": []
}
```

PowerShell

Untuk membatalkan Armada Reservasi Kapasitas

Gunakan Stop-EC2CapacityReservationFleet cmdlet.

```
Stop-EC2CapacityReservationFleet `
    -CapacityReservationFleetId crf-abcdef01234567890
```

Contoh konfigurasi Armada Reservasi Kapasitas

Contoh berikut membuat Armada Reservasi Kapasitas yang menggunakan dua tipe instans: `m5.4xlarge` dan `m5.12xlarge`.

Ini menggunakan sistem pembobotan berdasarkan jumlah vCPU yang disediakan oleh tipe instans yang ditentukan. Kapasitas target total adalah `480` vCPU. `m5.4xlarge` menyediakan 16 vCPU dan mempunyai bobot 16, sedangkan `m5.12xlarge` menyediakan 48 vCPU dan mempunyai bobot 48. Sistem pembobotan ini mengonfigurasi Armada Reservasi Kapasitas untuk cadangan kapasitas untuk 30 `m5.4xlarge` instans (480/16=30), atau 10 `m5.12xlarge` instans (480/48=10).

Armada dikonfigurasi untuk memprioritaskan kapasitas `m5.12xlarge` dan mendapatkan prioritas 1, sementara `m5.4xlarge` mendapatkan prioritas 2 yang lebih rendah. Ini berarti bahwa armada akan mencoba untuk memesan kapasitas `m5.12xlarge` terlebih dahulu, dan hanya mencoba untuk memesan kapasitas `m5.4xlarge` jika kapasitas `m5.12xlarge` Amazon EC2 tidak mencukupi.

Armada memesan kapasitas untuk instans `Windows` dan reservasi secara otomatis berakhir pada `October 31, 2021` pukul `23:59:59` UTC.

```
aws ec2 create-capacity-reservation-fleet \
    --total-target-capacity 480 \
    --allocation-strategy prioritized \
    --instance-match-criteria open \
    --tenancy default \
    --end-date 2021-10-31T23:59:59.000Z \
    --instance-type-specifications file://instanceTypeSpecification.json
```

Berikut ini adalah isi dari `instanceTypeSpecification.json`.

```
[
    {
        "InstanceType": "m5.4xlarge",
        "InstancePlatform":"Windows",
        "Weight": 16,
        "AvailabilityZone":"us-east-1a",
        "EbsOptimized": true,
```

```
        "Priority" : 2
    },
    {
        "InstanceType": "m5.12xlarge",
        "InstancePlatform":"Windows",
        "Weight": 48,
        "AvailabilityZone":"us-east-1a",
        "EbsOptimized": true,
        "Priority" : 1
    }
]
```

Menggunakan peran tertaut layanan untuk Armada Reservasi Kapasitas

On-Demand Armada Reservasi Kapasitas menggunakan AWS Identity and Access Management peran terkait layanan (I AM). Peran terkait layanan adalah jenis peran IAM unik yang ditautkan langsung ke Armada Reservasi Kapasitas. Service-linked peran telah ditentukan sebelumnya oleh Armada Reservasi Kapasitas dan menyertakan semua izin yang diperlukan layanan untuk memanggil AWS layanan lain atas nama Anda.

Peran terkait layanan membuat pengaturan Armada Reservasi Kapasitas menjadi lebih mudah karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. Armada Reservasi Kapasitas menentukan izin peran tertaut layanan, kecuali ditentukan lain, hanya Armada Reservasi Kapasitas yang dapat mengasumsikan perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, serta bahwa kebijakan izin tidak dapat dilampirkan ke entitas IAM lainnya.

Anda dapat menghapus peran tertaut layanan hanya setelah menghapus sumber daya terkait terlebih dahulu. Hal ini melindungi sumber daya Armada Reservasi Kapasitas karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

Service-linked izin peran untuk Armada Reservasi Kapasitas

Armada Reservasi Kapasitas menggunakan peran terkait layanan yang dinamai AWSServiceRoleForEC2CapacityReservationFleet untuk membuat, menjelaskan, memodifikasi, dan membatalkan Reservasi Kapasitas di Armada Reservasi Kapasitas atas nama Anda.

Per AWSServiceRoleForEC2CapacityReservationFleet an terkait layanan mempercayai entitas berikut untuk mengambil peran:

- `capacity-reservation-fleet.amazonaws.com`

Peran menggunakan kebijakan ter `AWSEC2CapacityReservationFleetRolePolicy` AWS kelola. Untuk informasi selengkapnya, lihat AWS kebijakan yang dikelola: AWSEC2CapacityReservationFleetRolePolicy.

Anda harus mengonfigurasi izin agar entitas IAM (seperti pengguna, grup, atau peran) dapat membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat izin Service-linked peran di Panduan Pengguna IAM.

Menggunakan peran tertaut layanan untuk Armada Reservasi Kapasitas

Anda tidak perlu membuat peran tertaut layanan secara manual. Saat Anda membuat Armada Reservasi Kapasitas menggunakan `create-capacity-reservation-fleet` AWS CLI perintah atau `CreateCapacityReservationFleet` API, peran terkait layanan secara otomatis dibuat untuk Anda.

Jika Anda menghapus peran tertaut layanan ini, dan ingin membuatnya lagi, Anda dapat mengulangi proses yang sama untuk membuat kembali peran tersebut di akun Anda. Saat Anda membuat Armada Reservasi Kapasitas, Armada Reservasi Kapasitas akan membuat peran terkait layanan untuk Anda lagi.

Mengedit peran tertaut layanan untuk Armada Reservasi Kapasitas

Armada Reservasi Kapasitas tidak mengizinkan Anda mengedit AWSServiceRoleForEC2CapacityReservationFleet peran terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin merujuk peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Meng edit deskripsi peran terkait layanan di Panduan Pengguna IAM.

Menghapus peran tertaut layanan untuk Armada Reservasi Kapasitas

Jika Anda tidak perlu lagi menggunakan fitur atau layanan yang memerlukan peran terkait layanan, kami merekomendasikan Anda menghapus peran tersebut. Dengan begitu, Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dipelihara secara aktif. Namun, Anda harus menghapus sumber daya untuk peran tertaut layanan sebelum menghapusnya secara manual.

> **Note**
>
> Jika layanan Armada Reservasi Kapasitas menggunakan peran tersebut saat Anda mencoba menghapus sumber daya, penghapusan mungkin gagal. Jika hal itu terjadi, tunggu beberapa menit dan coba mengoperasikannya lagi.

Untuk menghapus peran yang dit AWSServiceRoleForEC2CapacityReservationFleet autkan layanan

1. Gunakan `delete-capacity-reservation-fleet` AWS CLI perintah atau `DeleteCapacityReservationFleet` API untuk menghapus Armada Reservasi Kapasitas di akun Anda.
2. Gunakan konsol IAM, AWS CLI, atau AWS API untuk menghapus peran terkait AWSServiceRoleForEC2CapacityReservationFleet layanan. Untuk informasi selengkapnya, lihat Mengh apus peran terkait layanan di Panduan Pengguna IAM.

Wilayah yang Didukung untuk peran terkait layanan Armada Reservasi Kapasitas

Armada Reservasi Kapasitas mendukung penggunaan peran tertaut layanan di semua Wilayah tempat layanan tersedia. Untuk informasi selengkapnya, lihat Wilayah AWS dan Titik Akhir.

Memantau penggunaan Reservasi Kapasitas dengan CloudWatch metrik

Dengan CloudWatch metrik, Anda dapat memantau Reservasi Kapasitas secara efisien dan mengidentifikasi kapasitas yang tidak digunakan dengan menyet CloudWatch el alarm untuk memberi tahu Anda saat ambang batas penggunaan terpenuhi. Hal ini dapat membantu Anda mempertahankan volume Reservasi Kapasitas yang konstan dan mencapai tingkat pemanfaatan yang lebih tinggi.

Reservasi Kapasitas mengirimkan data metrik ke CloudWatch setiap lima menit. Metrik tidak didukung untuk Reservasi Kapasitas yang aktif kurang dari lima menit.

Untuk informasi selengkapnya tentang melihat metrik di CloudWatch konsol, lihat Menggunakan CloudWatch Metrik Amazon. Untuk informasi selengkapnya tentang membuat alarm, lihat Membuat CloudWatch Alarm Amazon.

Daftar Isi


- Metrik penggunaan Reservasi Kapasitas
- Dimensi metrik Reservasi Kapasitas
- Melihat CloudWatch metrik untuk Reservasi Kapasitas

### Metrik penggunaan Reservasi Kapasitas

Namespace `AWS/EC2CapacityReservations` mencakup metrik penggunaan berikut yang dapat Anda gunakan untuk memantau dan mempertahankan kapasitas sesuai permintaan dalam ambang batas yang Anda tentukan untuk reservasi Anda.

| Metrik | Deskripsi |
| --- | --- |
| `UsedInstanceCount` | Jumlah instans yang sedang digunakan.<br><br>Unit: Jumlah |
| `AvailableInstanceCount` | Jumlah instans yang tersedia.<br><br>Unit: Jumlah |
| `TotalInstanceCount` | Jumlah total instans yang telah Anda pesan.<br><br>Unit: Jumlah |
| `InstanceUtilization` | Persentase instans kapasitas terpesan yang saat ini sedang digunakan.<br><br>Satuan: Persen |

### Dimensi metrik Reservasi Kapasitas

Anda dapat menggunakan dimensi berikut untuk menyempurnakan metrik yang tercantum dalam tabel sebelumnya dalam Wilayah dan akun yang dipilih.

| Dimensi | Deskripsi |
| --- | --- |
| (Tidak ada dimensi) | Dimensi ini memfilter metrik yang ditentukan untuk semua Reservasi Kapasitas. |

| Dimensi | Deskripsi |
| --- | --- |
| `CapacityReservationId` | Dimensi ini menyaring metrik yang ditentukan untuk Reservasi Kapasitas yang diidentifikasi. |
| `InstanceType` | Dimensi ini menyaring metrik yang ditentukan untuk jenis instans yang diidentifikasi. |
| `AvailabilityZone` | Dimensi ini memfilter metrik yang ditentukan untuk Zona Ketersediaan yang diidentifikasi. |
| `InstanceMatchCriteria` | Dimensi ini memfilter metrik yang ditentukan untuk kriteria kecocokan instans yang diidentifikasi (`openatautargeted`). |
| `InstancePlatform` | Dimensi ini menyaring data metrik yang ditentukan untuk platform yang diidentifikasi. |
| `Tenancy` | Dimensi ini menyaring metrik yang ditentukan untuk sewa yang diidentifikasi. |

Melihat CloudWatch metrik untuk Reservasi Kapasitas

Metrik dikelompokkan berdasarkan namespace layanan, lalu dimensi yang didukung. Anda dapat menggunakan prosedur berikut untuk melihat metrik untuk Reservasi Kapasitas Anda.

Untuk melihat metrik Reservasi Kapasitas menggunakan CloudWatch konsol

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Jika perlu, ubah Wilayah. Dari bilah navigasi, pilih Wilayah tempat Anda Reservasi Kapasitas Anda berada. Untuk informasi selengkapnya, lihat Wilayah dan Titik Akhir.
3. Di panel navigasi, pilih Metrik.
4. Untuk Semua metrik, pilih Reservasi Kapasitas EC2.
5. Pilih dari dimensi metrik sebelumnya Di Semua Reservasi Kapasitas, Berdasarkan Reservasi Kapasitas, Berdasarkan Jenis Instans, Berdasarkan Zona Ketersediaan,

Berdasarkan Platform, Berdasarkan Kriteria Pencocokan Instans atau Berdasarkan Sewa dan metrik akan dikelompokkan berdasarkan Tidak ada dimensi,`CapacityReservationId`,`InstanceType`,,`AvailabilityZone`,`Platform`,`InstanceMat` dan `Tenancy` masing-masing.

6. Untuk mengurutkan metrik, gunakan judul kolom. Untuk membuat grafik metrik, pilih kotak centang di sebelah metrik.

Untuk melihat metrik Reservasi Kapasitas menggunakan AWS CLI

Gunakan perintah list-metrics berikut:

```
aws cloudwatch list-metrics --namespace "AWS/EC2CapacityReservations"
```

Memantau Pemanfaatan Pemesanan Kapasitas yang kurang

Anda dapat memantau kurangnya pemanfaatan Reservasi Kapasitas menggunakan cara berikut:

Topik

- Acara EventBridge Amazon
- Email dan AWS Health Pemberitahuan dasbor

Acara EventBridge Amazon

AWS Health mengirimkan acara ke Amazon EventBridge ketika Reservasi Kapasitas di akun Anda di bawah 20 persen penggunaan selama periode tertentu. Dengan EventBridge, Anda dapat menetapkan aturan yang memicu tindakan terprogram sebagai respons terhadap peristiwa tersebut. Misalnya, Anda dapat membuat aturan yang secara otomatis membatalkan Reservasi Kapasitas ketika pemanfaatannya turun di bawah 20 persen pemanfaatan selama periode 7 hari.

Peristiwa di EventBridge direpresentasikan sebagai objek JSON. Kolom-kolom yang unik untuk peristiwa tersebut terdapat di bagian "detail" dari objek JSON. Bidang "peristiwa" berisi nama peristiwa. Bidang "hasil" berisi status selesai dari tindakan yang memicu peristiwa. Untuk informasi selengkapnya, lihat pola EventBridge peristiwa Amazon di Panduan EventBridge Pengguna Amazon.

Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Fitur ini tidak didukung di AWS GovCloud (US).

Peristiwa

AWS Health mengirimkan peristiwa berikut ketika penggunaan kapasitas untuk Reservasi Kapasitas di bawah 20 persen.

- `AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION`

  Berikut ini adalah contoh peristiwa yang dihasilkan ketika Reservasi Kapasitas yang baru dibuat di bawah 20 persen penggunaan kapasitas selama periode 24 jam.

```
{
    "version": "0",
    "id": "b3e00086-f271-12a1-a36c-55e8ddaa130a",
    "detail-type": "AWS Health Event",
    "source": "aws.health",
    "account": "123456789012",
    "time": "2023-03-10T12:03:38Z",
    "region": "ap-south-1",
    "resources": [
        "cr-01234567890abcdef"
    ],
    "detail": {
        "eventArn": "arn:aws:health:ap-south-1::event/EC2/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_cr-01234567890abcdef-6211-4d50-9286-0c9fbc243f04",
        "service": "EC2",
        "eventTypeCode": "AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION",
        "eventTypeCategory": "accountNotification",
        "startTime": "Fri, 10 Mar 2023 12:03:38 GMT",
        "endTime": "Fri, 10 Mar 2023 12:03:38 GMT",
        "eventDescription": [
            {
                "language": "en_US",
                "latestDescription": "A description of the event will be provided
 here"
            }
        ],
        "affectedEntities": [
            {
                "entityValue": "cr-01234567890abcdef"
            }
        ]
    }
```

```
    }
```

- AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY

  Berikut ini adalah contoh peristiwa yang dihasilkan ketika satu atau lebih Reservasi Kapasitas di bawah 20 persen penggunaan kapasitas selama periode 7 hari.

```
{
    "version": "0", "id":"7439d42b-3c7f-ad50-6a88-25e2a70977e2",
    "detail-type": "AWS Health Event",
    "source": "aws.health",
    "account": "123456789012",
    "time": "2023-03-07T06:06:01Z",
    "region": "us-east-1",
    "resources": [
        "cr-01234567890abcdef | us-east-1b | t3.medium | Linux/UNIX | 0.0%",
        "cr-09876543210fedcba | us-east-1a | t3.medium | Linux/UNIX | 0.0%"
    ],
    "detail": {
        "eventArn": "arn:aws:health:us-east-1::event/
EC2/AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY/
AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY_726c1732-d6f6-4037-b9b8-
bec3c2d3ba65",
        "service": "EC2",
        "eventTypeCode": "AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION_SUMMARY",
        "eventTypeCategory": "accountNotification",
        "startTime": "Tue, 7 Mar 2023 06:06:01 GMT",
        "endTime": "Tue, 7 Mar 2023 06:06:01 GMT",
        "eventDescription": [
            {
                "language": "en_US",
                "latestDescription": "A description of the event will be provided
 here"
            }
        ],
        "affectedEntities": [
            {
                "entityValue": "cr-01234567890abcdef | us-east-1b | t3.medium |
 Linux/UNIX | 0.0%"
            },
            {
                "entityValue": "cr-09876543210fedcba | us-east-1a | t3.medium |
 Linux/UNIX | 0.0%"
```

```
            }
        ]
    }
}
```

### Buat EventBridge aturan

Untuk menerima pemberitahuan email saat pemanfaatan Reservasi Kapasitas turun di bawah 20 persen, buat topik Amazon SNS, lalu buat EventBridge aturan untuk `AWS_EC2_ODCR_UNDERUTILIZATION_NOTIFICATION` acara tersebut.

Untuk membuat topik Amazon SNS

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Pada panel navigasi, silakan pilih Topik, lalu pilih Buat topik.
3. Untuk Tipe, pilih Standar.
4. Untuk Nama, masukkan nama untuk topik baru.
5. Pilih Buat topik.
6. Pilih Buat langganan.
7. Untuk Protokol, pilih Email, lalu untuk Titik akhir, masukkan alamat email yang menerima notifikasi.
8. Pilih Buat langganan.
9. Alamat email yang dimasukkan di atas akan menerima pesan email dengan baris subjek berikut: `AWS Notification - Subscription Confirmation`. Ikuti petunjuk untuk mengonfirmasi langganan Anda.

Untuk membuat EventBridge aturan

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Di panel navigasi, pilih Aturan, lalu pilih Buat aturan.
3. Untuk Nama, masukkan nama untuk aturan baru.
4. Untuk Tipe aturan, pilih Aturan dengan pola peristiwa.
5. Pilih Selanjutnya.
6. Untuk Pola peristiwa, lakukan hal berikut:

a. Untuk Sumber peristiwa, pilih Layanan AWS .
b. Untuk Layanan AWS , pilih AWS Health.
c. Untuk Tipe peristiwa, pilih EC2 ODCR Underutilization Notification.

7. Pilih Selanjutnya.
8. Untuk Target 1, lakukan hal berikut:

a. Untuk Tipe Target, pilih Layanan AWS .
b. Untuk Pilih target, pilih Topik SNS.
c. Untuk Topic, pilih topik yang Anda buat sebelumnya.

9. Pilih Berikutnya lalu Berikutnya lagi.
10. Pilih Buat aturan.

### Email dan AWS Health Pemberitahuan dasbor

AWS Health mengirimkan email dan Dasbor Health pemberitahuan berikut ketika pemanfaatan kapasitas untuk Reservasi Kapasitas di akun Anda turun di bawah 20 persen.

- Notifikasi individual untuk setiap Reservasi Kapasitas yang baru dibuat dengan pemanfaatan di bawah 20 persen selama periode 24 jam terakhir.
- Ringkasan notifikasi untuk semua Reservasi Kapasitas dengan pemanfaatan di bawah 20 persen selama periode 7 hari terakhir.

Pemberitahuan dan Dasbor Health pemberitahuan email dikirim ke alamat email yang terkait dengan AWS akun yang memiliki Reservasi Kapasitas. Notifikasi mencakup informasi berikut:

- ID Reservasi Kapasitas.
- Zona Ketersediaan dari Reservasi Kapasitas.
- Tingkat pemanfaatan rata-rata untuk Reservasi Kapasitas.
- Tipe instans dan platform (sistem operasi) dari Reservasi Kapasitas.

Selain itu, ketika pemanfaatan kapasitas untuk Reservasi Kapasitas di akun Anda turun di bawah 20 persen selama periode 24 jam dan 7 hari, AWS Health mengirimkan acara ke EventBridge. DenganEventBridge, Anda dapat membuat aturan yang mengaktifkan tindakan otomatis, seperti mengirim pemberitahuan email atau AWS Lambda fungsi memicu, sebagai respons terhadap

peristiwa tersebut. Untuk informasi selengkapnya, lihat Memantau Pemanfaatan Pemesanan Kapasitas yang kurang.

Pantau perubahan status untuk Reservasi Kapasitas di masa mendatang

Amazon EC2 mengirimkan acara ke Amazon EventBridge ketika status Reservasi Kapasitas di masa mendatang berubah.

Berikut ini adalah contoh dari acara ini. Dalam contoh ini, Reservasi Kapasitas tanggal mendatang memasuki negara bagian`scheduled`. Perhatikan status yang disorot di `detail-type` lapangan.

```
{
   "version":"0",
   "id":"12345678-1234-1234-1234-123456789012",
   "detail-type":"EC2 Capacity Reservation Scheduled",
   "source":"aws.ec2",
   "account":"123456789012",
   "time":"yyyy-mm-ddThh:mm:ssZ",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-1234567890abcdefg"
   ],
   "detail":{
      "capacity-reservation-id":"cr-1234567890abcdefg",
      "state":"scheduled"
   }
}
```

Nilai yang mungkin untuk `detail-type` bidang ini adalah:

- `Scheduled`
- `Active`
- `Delayed`
- `Unsupported`
- `Failed`
- `Expired`

Untuk informasi selengkapnya tentang status ini, lihat Melihat status Reservasi Kapasitas.

Anda dapat membuat EventBridge acara Amazon yang memantau peristiwa ini dan kemudian memicu tindakan tertentu saat terjadi. Untuk informasi selengkapnya, lihat Membuat aturan yang bereaksi terhadap peristiwa di Amazon EventBridge.

Untuk membuat aturan yang memantau semua peristiwa perubahan status, Anda dapat menggunakan pola peristiwa berikut.

```
{
  "source": ["aws.ec2"],
  "detail-type": [{
    "prefix": "EC2 Capacity Reservation"
  }]
}
```

Untuk membuat aturan yang hanya memantau perubahan status tertentu, Anda dapat menggunakan pola peristiwa berikut.

```
{
  "source": ["aws.ec2"],
  "detail-type": [{
    "prefix": "EC2 Capacity Reservation state"
  }]
}
```

Misalnya, pola peristiwa berikut memantau peristiwa yang dikirim saat Reservasi Kapasitas tanggal mendatang memasuki `active` status.

```
{
  "source": ["aws.ec2"],
  "detail-type": [{
    "prefix": "EC2 Capacity Reservation Active"
  }]
}
```

### Reservasi Kapasitas yang Dapat Diinterupsi

Reservasi Kapasitas Terputus membantu Anda membuat kapasitas yang tidak terpakai tersedia sementara untuk beban kerja lain dalam akun Anda. Ini memberi Anda kontrol untuk merebut kembali kapasitas saat dibutuhkan. Saat Anda merebut kembali kapasitas, instans apa pun yang berjalan di dalam reservasi yang dapat diinterupsi dihentikan. Setelah membuat reservasi yang dapat diinterupsi,

Anda dapat membagikannya dengan AWS akun tambahan atau AWS organisasi Anda menggunakan AWS Resource Access Manager (RAM).

Gunakan Reservasi Kapasitas yang dapat diinterupsi saat Anda memiliki kapasitas cadangan yang tidak terpakai selama periode di luar puncak, di antara penerapan, atau saat beban kerja menurun. Jika Anda tahu tim lain yang dapat menggunakan kapasitas ini, Anda dapat membuatnya tersedia dengan membuat Reservasi Kapasitas yang dapat diinterupsi. Ketika beban kerja kritis Anda membutuhkan kapasitas kembali, Anda dapat merebutnya kembali.

Anda dapat menggunakan Reservasi Kapasitas yang dapat diinterupsi sebagai salah satu dari berikut ini:

- Pemilik Kapasitas — Anda memiliki Reservasi Kapasitas sumber dan membuat Reservasi Kapasitas yang dapat diinterupsi untuk berbagi kapasitas yang tidak terpakai dengan tim lain sambil mempertahankan kontrol untuk merebutnya kembali bila diperlukan.
- Konsumen kapasitas — Anda meluncurkan instans ke reservasi terputus bersama, memahami bahwa instans Anda mungkin dihentikan ketika pemilik mengklaim kembali kapasitas.

Topik


- Cara kerjanya
- Penagihan
- Pertimbangan-pertimbangan
- Reservasi Kapasitas yang Dapat Diinterupsi untuk pemilik kapasitas
- Reservasi Kapasitas yang Dapat Diinterupsi untuk konsumen kapasitas
- Memantau Reservasi Kapasitas yang dapat terputus dengan dan EventBridge CloudTrail


## Cara kerjanya

Untuk membuat kapasitas yang tidak terpakai tersedia bagi tim lain, buat reservasi yang dapat diinterupsi dengan menentukan jumlah instans yang tidak digunakan yang ingin Anda bagikan dari reservasi sumber Anda. Saat Anda membuat reservasi yang dapat diinterupsi, kami mentransfer instans ini dari reservasi sumber Anda ke reservasi baru yang dapat diinterupsi di akun Anda.

Kami mempertahankan hubungan antara reservasi sumber dan Reservasi Kapasitas yang dapat diinterupsi. Akibatnya, saat Anda merebut kembali kapasitas Anda, instans konsumen yang sedang berjalan dihentikan, dan kapasitas dikembalikan ke reservasi sumber asli Anda.

Fitur kunci:

- Jadikan kapasitas yang tidak terpakai tersedia sementara sambil mempertahankan kontrol untuk merebutnya kembali
- Reklam kembali kapasitas kapan saja. Untuk informasi selengkapnya, lihat Proses reklamasi dan pelacakan
- Berbagi dengan akun tambahan atau AWS organisasi Anda menggunakan AWS Resource Access Manager (RAM)

Penagihan

Saat membuat reservasi yang dapat diinterupsi, Anda akan ditagih sebagai reservasi baru yang independen. Ini membagi tagihan Anda:

- Reservasi sumber: Anda ditagih untuk total kapasitas dikurangi kapasitas yang dialokasikan
- Reservasi yang dapat diinterupsi: Anda ditagih untuk kapasitas yang dialokasikan

Untuk informasi selengkapnya tentang penagihan Reservasi On-Demand Kapasitas, lihat Harga dan penagihan Reservasi Kapasitas.

Pertimbangan-pertimbangan

Sebelum menggunakan Reservasi Kapasitas yang dapat diinterupsi, tinjau batasan dan persyaratan berikut yang berlaku untuk pemilik kapasitas dan konsumen.

Pemilik kapasitas

- Anda tidak dapat secara langsung mengubah atau membatalkan Reservasi Kapasitas yang dapat diinterupsi. Untuk mengeditnya, perbarui kapasitas yang dialokasikan dari sumber Reservasi Kapasitas.
- Anda hanya dapat melihat, meluncurkan, memberi tag, berbagi, dan menetapkan tagihan untuk reservasi yang dapat diinterupsi.
- Anda hanya dapat membuat satu alokasi yang dapat diinterupsi per sumber Reservasi Kapasitas.

Konsumen kapasitas

- Reservasi Kapasitas yang Dapat Diinterupsi secara default adalah Reservasi Kapasitas yang ditargetkan, jadi Anda perlu menargetkannya saat peluncuran instans.

- Anda tidak dapat menambahkan Reservasi Kapasitas yang dapat diinterupsi ke grup reservasi kapasitas.
- Kami menyarankan Anda hanya menggunakan Reservasi Kapasitas yang dapat terputus untuk aplikasi yang dapat terganggu.
- Instans Anda akan dihentikan ketika pemilik mengklaim kembali kapasitas - tidak ada fallback ke On-Demand atau Spot. Untuk informasi selengkapnya, lihat Pengalaman interupsi.

## Reservasi Kapasitas yang Dapat Diinterupsi untuk pemilik kapasitas

Pemilik kapasitas adalah akun yang memiliki Reservasi Kapasitas sumber dan membuat Reservasi Kapasitas yang dapat diinterupsi untuk berbagi kapasitas yang tidak terpakai dengan tim lain sambil mempertahankan kontrol untuk merebutnya kembali bila diperlukan.

Bagian ini mencakup bagaimana Anda (pemilik kapasitas) dapat membuat, memodifikasi, mengklaim kembali, dan melacak Reservasi Kapasitas yang dapat diinterupsi.

Topik

- Membuat Reservasi Kapasitas yang Dapat Diinterupsi
- Lihat Reservasi Kapasitas yang dapat diinterupsi
- Memodifikasi Reservasi Kapasitas yang dapat diinterupsi
- Mempertahankan Reservasi Kapasitas yang Dapat Diinterupsi pada kapasitas nol
- Proses reklamasi dan pelacakan
- Berbagi reservasi yang dapat diinterupsi

## Membuat Reservasi Kapasitas yang Dapat Diinterupsi

Buat Reservasi Kapasitas yang dapat diinterupsi untuk membuat kapasitas yang tidak terpakai dari reservasi sumber Anda tersedia untuk beban kerja lain sambil mempertahankan kontrol untuk merebutnya kembali saat diperlukan.

### Prasyarat

Sebelum membuat alokasi yang dapat diinterupsi, pastikan Reservasi On-Demand Kapasitas sumber Anda memenuhi persyaratan berikut:

- Reservasi Kapasitas Anda harus dalam keadaan aktif tanpa tanggal akhir ditetapkan. Anda tidak dapat membuat alokasi dari reservasi yang tertunda, kedaluwarsa, dibatalkan, atau memiliki tanggal akhir yang dijadwalkan.
- Reservasi Kapasitas Anda harus memiliki kapasitas yang tersedia untuk alokasi. Anda hanya dapat mengalokasikan instans yang tersedia (juga disebut kapasitas yang tidak digunakan).
- Anda hanya dapat membuat satu alokasi yang dapat diinterupsi per sumber Reservasi Kapasitas. Jika alokasi sudah ada, Anda harus memodifikasi atau membatalkannya sebelum membuat yang baru.
- Anda dapat mengalokasikan maksimal 1000 instans sekaligus ke Reservasi Kapasitas yang dapat diinterupsi.

Anda dapat menggunakan konsol atau AWS CLI untuk membuat Reservasi Kapasitas yang dapat diinterupsi.

Console

Untuk membuat Reservasi Kapasitas yang dapat diinterupsi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Reserv asi Kapasitas.
3. Pilih Reservasi Kapasitas Anda.
4. Pilih Tind akan, Buat reservasi yang dapat diinterupsi.
5. Untuk Kuantitas, masukkan jumlah instance yang akan dialokasikan.
6. Untuk preferensi Alokasi saat semua kapasitas direklamasi, pilih Def ault atau Per tahankan.
7. (Opsional) Tambahkan tanda.
8. Pilih Buat reservasi yang dapat diinterupsi.

AWS CLI

Untuk membuat Reservasi Kapasitas yang dapat diinterupsi

Gunakan perintah create-interruptible-capacity-reservation-:

```
aws ec2 create-interruptible-capacity-reservation-allocation \
    --capacity-reservation-id cr-1234567890abcdef0 \
```

```
    --instance-count 10
```

Lihat Reservasi Kapasitas yang dapat diinterupsi

Setelah membuat Reservasi Kapasitas yang dapat diinterupsi, Anda dapat melihat reservasi yang dapat diinterupsi di akun Anda atau dari sumber daya tertentu.

Lihat semua Reservasi Kapasitas yang dapat diinterupsi di akun Anda

Gunakan prosedur berikut untuk melihat Reservasi Kapasitas yang dapat diinterupsi di akun Anda.

Console

Untuk melihat Reservasi Kapasitas yang dapat diinterupsi di akun Anda

1. Buka halaman Reservasi Kapasitas di konsol.
2. Di kolom Interruptible, cari reservasi bertanda Ya.
3. Pilih reservasi yang dapat diinterupsi untuk melihat detail.

AWS CLI

Untuk melihat Reservasi Kapasitas yang dapat diinterupsi di akun Anda

```
aws ec2 describe-capacity-reservations \
    --capacity-reservation-id cr-interruptible-id \
    --filters Name=interruptible,Values=true
```

Melihat Reservasi Kapasitas yang dapat diinterupsi dari sumber tertentu

Gunakan prosedur berikut untuk melihat Reservasi Kapasitas yang dapat diinterupsi yang dibuat dari sumber tertentu Reservasi Kapasitas.

```
aws ec2 describe-capacity-reservations \
    --capacity-reservation-id cr-source-id
```

Dalam tanggapan, Anda akan menemukan `interruptibleCapacityAllocation` objek yang berisi ID Reservasi Kapasitas yang dapat diinterupsi dan detail alokasi. Untuk informasi tentang struktur respons, lihat InterruptibleCapacityAllocation di Refer ensi API Amazon EC2.

## Memodifikasi Reservasi Kapasitas yang dapat diinterupsi

Gunakan prosedur berikut untuk mengedit atau membatalkan Reservasi Kapasitas yang dapat diinterupsi.

> **Note**
>
> - Saat Anda mengurangi alokasi, pertama-tama kami mengklaim kembali instans yang tersedia, kemudian menjalankan instance, hingga kami memenuhi jumlah yang diminta. Jika kami dapat memenuhi jumlah sepenuhnya dengan instance yang tersedia, tidak ada penghentian yang terjadi. Semua modifikasi pada jumlah instans yang dialokasikan dilakukan melalui Reservasi Kapasitas sumber, bukan langsung pada Reservasi Kapasitas yang dapat diinterupsi.
> - Anda hanya dapat mengubah Reservasi Kapasitas yang dapat diinterupsi dengan maksimum 1000 instans sekaligus (menambah atau mengurangi).

### Mengedit Reservasi Kapasitas yang dapat diinterupsi

Gunakan prosedur berikut untuk mengedit Reservasi Kapasitas yang dapat diinterupsi.

#### Console

1. Dari halaman rincian Reservasi Kapasitas sumber, pilih  tautan  Edit alokasi yang dapat diinterupsi.
2. Untuk  Kuantitas, masukkan nomor baru:
   - Tambahkan lebih banyak kapasitas untuk dibagikan
   - Dapatkan kembali kapasitas ke sumber Anda Reservasi Kapasitas
3. Pilih Perbarui.

#### AWS CLI

```
aws ec2 update-interruptible-capacity-reservation-allocation \
    --capacity-reservation-id cr-1234567890abcdef0 \
    --target-instance-count 80
```

### Batalkan Reservasi Kapasitas yang Dapat Diinterupsi

Gunakan prosedur berikut untuk menghapus alokasi secara permanen dan mengembalikan semua kapasitas.

Console

1. Dari halaman rincian Reservasi Kapasitas sumber, navigasikan ke detail alokasi kapasitas yang dapat diinterupsi.
2. Pilih  Edit alokasi  yang dapat diinterupsi.
3. Untuk Kuantitas , masukkan`0` .
4. Pilih Perbarui.

AWS CLI

```
aws ec2 update-interruptible-capacity-reservation-allocation \
--capacity-reservation-id cr-1234567890abcdef0 \
--target-instance-count 0
```

> **Pertahankan preferensi ukuran nol**
>
> Jika Anda mengonfigurasi `retain` untuk preferensi ukuran nol, Reservasi Kapasitas yang dapat diinterupsi tetap aktif pada instans nol. Untuk membatalkannya, perbarui preferensi ukuran nol ke`default`, lalu kurangi alokasi menjadi nol. Untuk informasi selengkapnya, lihat Mempertahankan Reservasi Kapasitas yang Dapat Diinterupsi pada kapasitas nol.

### Mempertahankan Reservasi Kapasitas yang Dapat Diinterupsi pada kapasitas nol

Secara default, Amazon EC2 membatalkan reservasi yang dapat diinterupsi saat Anda mengurangi alokasinya menjadi nol instans. Kapasitas yang direklamasi kembali ke Reservasi Kapasitas sumber Anda.

Jika Anda berharap untuk berbagi kapasitas lagi nanti, Anda dapat mengonfigurasi reservasi yang dapat diinterupsi agar tetap aktif pada instans nol. Dengan opsi ini, Anda dapat mengalokasikan instance ke reservasi yang sama nanti tanpa membuatnya ulang.

Anda mengontrol perilaku ini dengan preferensi ukuran nol. Anda dapat mengaturnya saat membuat atau mengedit alokasi Reservasi Kapasitas yang dapat diinterupsi, baik di konsol atau dengan. AWS CLI

Console

Pada halaman Buat reservasi yang dapat diinterupsi atau Edit alokasi yang dapat diinterupsi, untuk preferensi Alokasi saat semua kapasitas direklamasi, pilih salah satu dari yang berikut ini:

- Default — Amazon EC2 membatalkan Reservasi Kapasitas yang dapat diinterupsi saat Anda mengurangi alokasinya menjadi nol.
- Pertahankan — Reservasi Kapasitas yang dapat diinterupsi tetap aktif pada kapasitas nol saat Anda mengurangi alokasinya menjadi nol. Anda kemudian dapat mengalokasikan instance untuk itu lagi nanti.

Setelah Anda menyimpan perubahan, pengaturan akan muncul sebagai bidang preferensi ukuran Nol yang hanya dapat dibaca pada detail Reservasi Kapasitas yang dapat diinterupsi.

AWS CLI

Untuk menjaga Reservasi Kapasitas yang dapat diinterupsi tetap aktif pada kapasitas nol, tentukan `--zero-size-preference retain` kapan Anda memanggil alokasi create-interruptible-capacity-reservation-atau update-interruptible-capacity-reservation-alocation-. https://awscli.amazonaws.com/v2/documentation/api/latest/reference/ec2/update-interruptible-capacity-reservation-allocation.html Untuk membatalkan reservasi ketika alokasinya mencapai nol, tentukan`default`.

```
aws ec2 update-interruptible-capacity-reservation-allocation \
    --capacity-reservation-id cr-1234567890abcdef0 \
    --zero-size-preference retain
```

Proses reklamasi dan pelacakan

Ketika Anda merebut kembali kapasitas:

- Instans yang sedang berjalan menerima peringatan gangguan selama 2 menit melalui EventBridge peristiwa.
- Setelah periode pemberitahuan, instans yang berjalan dalam kapasitas reklamasi memasuki status shutdown dan dihentikan.

- Saat dihentikan, instans yang direklamasi akan tersedia di Reservasi Kapasitas sumber Anda untuk segera digunakan.
- Status alokasi Anda berubah dari pembaruan menjadi aktif setelah selesai.

Reklamasi lengkap dapat memakan waktu beberapa menit tergantung pada jenis instans dan waktu shutdown. Untuk informasi selengkapnya tentang EventBridge notifikasi yang Anda terima saat proses selesai, lihatPenyelesaian reklamasi.

Lacak status reklamasi

Pantau kemajuan reklamasi dengan menjelaskan reservasi sumber Anda:

```
aws ec2 describe-capacity-reservations \
--capacity-reservation-id cr-1234567890abcdef0
```

Respons menunjukkan bidang-bidang ini di dalam `interruptibleCapacityAllocation` objek:

- `instance-count`: Instans yang dialokasikan saat ini
- `target-instance-count`: Kuantitas yang diminta setelah reklamasi
- `status`: memperbarui selama reklamasi dan aktif saat selesai

Berbagi reservasi yang dapat diinterupsi

Anda dapat membagikan reservasi yang dapat diinterupsi hanya dalam AWS organisasi Anda menggunakan AWS Resource Access Manager (RAM).

Pertimbangan:

- Jika akun konsumen meninggalkan organisasi Anda, reservasi yang dapat diinterupsi secara otomatis tidak dibagikan dari akun tersebut.
- Instans apa pun yang berjalan di reservasi yang tidak dibagikan akhirnya dihentikan.
- Semua fungsi berbagi lainnya berfungsi sama dengan Reservasi Kapasitas standar.

Untuk prosedur berbagi lengkap, lihat Pemesanan Kapasitas Berbagi.

## Reservasi Kapasitas yang Dapat Diinterupsi untuk konsumen kapasitas

Konsumen kapasitas adalah akun yang meluncurkan instans ke Reservasi Kapasitas yang dapat diinterupsi bersama, memahami bahwa instans mereka dapat dihentikan ketika pemilik mengklaim kembali kapasitas.

Bagian ini membahas bagaimana Anda (konsumen kapasitas) dapat meluncurkan instans ke Reservasi Kapasitas yang dapat diinterupsi dan mempelajari tentang apa yang terjadi ketika kapasitas direklamasi kembali oleh pemilik.

Topik

- Melihat Reservasi Kapasitas yang dapat diinterupsi
- Luncurkan instance ke reservasi yang dapat diinterupsi
- Pengalaman interupsi

### Melihat Reservasi Kapasitas yang dapat diinterupsi

Gunakan prosedur berikut untuk melihat Reservasi Kapasitas yang dapat diinterupsi.

Console

Untuk melihat Reservasi Kapasitas yang dapat diinterupsi di akun Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Reserv asi Kapasitas.
3. Di kolom Interruptible, cari reservasi bertanda Ya.
4. Perhatikan ID reservasi untuk digunakan dalam peluncuran instans Anda.

AWS CLI

Untuk menemukan semua Reservasi Kapasitas yang dapat diinterupsi di akun Anda

Gunakan perintah describe-capacity-reservations:

```
aws ec2 describe-capacity-reservations \
--filters Name=state,Values=active
```

Cari reservasi di mana `Interruptible` diatur `true` dalam tanggapan.

Untuk memfilter khusus untuk reservasi yang dapat diinterupsi

Gunakan perintah berikut ini.

```
aws ec2 describe-capacity-reservations \
--capacity-reservation-ids cr-example123 \
--query 'CapacityReservations[?Interruptible==`true`]'
```

> **Note**
>
> Reservasi Kapasitas yang Dapat Diinterupsi secara default adalah Reservasi Kapasitas yang ditargetkan, jadi Anda perlu menargetkannya secara khusus dalam peluncuran instans Anda. Tidak seperti reservasi terbuka, reservasi yang dapat diinterupsi tidak akan secara otomatis mencakup instans yang cocok. Anda harus secara eksplisit menentukan ID reservasi saat meluncurkan.

Luncurkan instance ke reservasi yang dapat diinterupsi

Gunakan prosedur berikut untuk meluncurkan instans Amazon EC2 ke dalam Reservasi Kapasitas yang dapat diinterupsi dalam akun Anda.

> **Note**
>
> Kami menyarankan Anda hanya menggunakan Reservasi Kapasitas yang dapat terputus untuk aplikasi yang dapat terganggu.

Console

Untuk meluncurkan instans ke Reservasi Kapasitas yang dapat diinterupsi

1. Buka konsol Amazon Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari dasbor Amazon EC2, pilih Luncurkan Instans.
3. Konfigurasikan setelan instans Anda.
4. Di Detail lanjutan untuk Reservasi Kapasitas, pilih Luncurkan instans yang dapat diinterupsi di reserv asi aktif Anda.

5. Pilih ID reservasi yang dapat diinterupsi dan opsi pembelian instans baru.
6. Pilih Luncurkan instans.

AWS CLI

```
aws ec2 run-instances \
--instance-type m5.large \
--count 2 \
--image-id ami-12345678 \
--instance-market-options '{
    "MarketType": "interruptible-capacity-reservation"
}' \
--capacity-reservation-specification '{
    "CapacityReservationTarget": {
        "CapacityReservationId": "cr-abcdef1234567890"
    }
}'
```

Luncurkan instans dengan Grup Penskalaan Otomatis

Anda juga dapat meluncurkan instans ke reservasi yang dapat diinterupsi menggunakan Grup Penskalaan Otomatis dengan templat peluncuran. Konfigurasikan template peluncuran dengan tipe pasar yang dapat diinterupsi dan ID reservasi, lalu buat Grup Penskalaan Otomatis menggunakan templat tersebut. Untuk informasi selengkapnya, lihat Reservasi Kapasitas Terputus dengan Penskalaan Otomatis EC2.

Pengalaman interupsi

Ketika kapasitas direklamasi kembali oleh pemilik, Anda menerima pemberitahuan gangguan 2 menit sebelum penghentian instans. Peringatan ini datang melalui EventBridge peristiwa, memberi Anda waktu untuk:

- Simpan pekerjaan Anda atau periksa aplikasi Anda
- Matikan proses
- Persiapkan misalnya penghentian

Acara EventBridge ini mencakup rincian tentang instans mana yang akan dihentikan dan waktu penghentian yang tepat. Untuk informasi selengkapnya, lihat Peringatan gangguan instans.

Memantau Reservasi Kapasitas yang dapat terputus dengan dan EventBridge CloudTrail

Reservasi Kapasitas yang Dapat Di EventBridge interupsi mengirimkan pemberitahuan dan CloudTrail acara untuk membantu Anda memantau dan menanggapi perubahan kapasitas.

Topik

- EventBridge notifikasi
- CloudTrail acara

### EventBridge notifikasi

Anda menerima dua jenis EventBridge notifikasi. Untuk informasi tentang cara mengatur EventBridge notifikasi, lihat Membuat EventBridge aturan Amazon.

#### Peringatan gangguan instans

Jika menjalankan instans dalam reservasi yang dapat diinterupsi, Anda menerima pemberitahuan ini 2 menit sebelum instans dihentikan:

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "EC2 Capacity Reservation Instance Interruption Warning",
    "source": "aws.ec2",
    "account": "[instance owner Account ID]",
    "time": "[Current time in yyyy-mm-ddThh:mm:ssZ]",
    "resources": "[instance arn]",
    "region": "[region]",
    "detail": {
        "instance-id": "[instance-id]",
        "instance-action": "terminate",
        "instance-termination-time": "yyyy-mm-ddThh:mm:ssZ",
        "azId": "[availability-zone-id]"
    }
}
```

#### Penyelesaian reklamasi

Jika Anda memiliki reservasi sumber, Anda menerima pemberitahuan ini ketika reklamasi kapasitas selesai:

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail-type": "EC2 Interruptible Capacity Reservation Allocation Reclamation
 Completed",
    "source": "aws.ec2",
    "account": "[source Capacity Reservation Owner Account ID]",
    "time": "[Current time in yyyy-mm-ddThh:mm:ssZ]",
    "region": "us-east-1",
    "resources": ["source_cr_arn"],
    "detail": {
        "sourceCapacityReservationId": "string",
        "instanceType": "string",
        "availabilityZoneId": "string",
        "TotalInstanceCount": "current total count in the source",
        "ReclaimedInstanceCount": "count of instances added to the source",
        "targetInstanceCount": "number"
    }
}
```

CloudTrail acara

CloudTrail mencatat peristiwa ini untuk Reservasi Kapasitas yang dapat diinterupsi:

- `InterruptibleCapacityReservationCreated`— Saat Anda membuat alokasi yang dapat diinterupsi
- `InterruptibleCapacityReservationAllocationUpdated`— Saat Anda memodifikasi alokasi
- `InterruptibleCapacityReservationCancelled`— Saat Anda membatalkan alokasi
- `CapacityReservationModified`— Saat kami memodifikasi reservasi sumber untuk alokasi
- `InterruptibleCapacityReservationInstancesTerminated`— Ketika kami menghentikan instans selama reklamasi

## Blok Kapasitas untuk ML

Blok Kapasitas untuk ML memungkinkan Anda untuk memesan inst GPU-based ans komputasi yang dipercepat pada tanggal mendatang untuk mendukung beban kerja pembelajaran mesin (ML) durasi pendek Anda. Instans yang berjalan di dalam Blok Kapasitas secara otomatis ditempatkan

berdekatan di dalam Amazon EC2 UltraClusters, untuk jaringan latensi rendah, skala petabit, dan non-pemblokiran.

Anda juga dapat menggunakan Blok Kapasitas untuk mencadangkan kapasitas untuk Amazon EC2 UltraServers. UltraServers hubungkan beberapa instans Amazon EC2 dalam interkoneksi akselerator bandwidth tinggi latensi rendah. Anda dapat menggunakannya UltraServers untuk menangani AI/ML beban kerja komputasi dan memori yang paling intensif dalam pelatihan, penyempurnaan, dan inferensi. Untuk informasi selengkapnya, lihat Amazon EC2 UltraServers.

Dengan Blok Kapasitas, Anda dapat melihat kapan kapasitas instans GPU tersedia di masa mendatang, dan Anda dapat menjadwalkan Blok Kapasitas untuk memulai pada waktu yang paling sesuai untuk Anda. Saat Anda memesan Blok Kapasitas, Anda mendapatkan jaminan kapasitas yang dapat diprediksi untuk instans GPU dengan membayar jumlah waktu yang Anda butuhkan saja. Kami merekomendasikan Blok Kapasitas saat Anda membutuhkan GPU untuk mendukung beban kerja ML Anda selama berhari-hari atau berminggu-minggu sekaligus dan tidak ingin membayar reservasi saat instans GPU Anda tidak digunakan.

Berikut ini adalah beberapa kasus penggunaan umum untuk Blok Kapasitas.

- Pelatihan model ML dan fine-tuning — Dapatkan akses tanpa gangguan ke instans GPU yang Anda pesan untuk menyelesaikan pelatihan model dan fine-tuning.
- Eksperimen dan prototipe ML — Jalankan eksperimen dan bangun prototipe yang memerlukan instans GPU untuk jangka waktu pendek.

Blok Kapasitas tersedia untuk jenis instans tertentu di beberapa Wil AWS ayah. Untuk informasi selengkapnya, lihat Jenis dan Wilayah instans yang didukung.

Anda dapat memesan Blok Kapasitas dengan waktu mulai reservasi hingga delapan minggu ke depan. Setiap Blok Kapasitas dapat memiliki hingga 64 instans, dan Anda dapat memiliki hingga 256 instans di seluruh Blok Kapasitas.

Topik


- Jenis dan Wilayah instans yang didukung
- Platform yang didukung
- Pertimbangan-pertimbangan
- Sumber daya terkait
- Cara kerja Blok Kapasitas Amazon EC2

- Harga dan penagihan Blok Kapasitas
- Temukan dan beli Blok Kapasitas
- Luncurkan instans menggunakan Blok Kapasitas
- Melihat Blok Kapasitas
- Perluas Blok Kapasitas
- Bagikan Blok Kapasitas
- Membuat Grup Sumber Daya Reservasi Kapas UltraServer itas untuk Blok Kapasitas
- Memantau Blok Kapasitas menggunakan EventBridge
- Pencatatan Kapasitas Memblokir panggilan API dengan AWS CloudTrail

### Jenis dan Wilayah instans yang didukung

Blok Instance dan UltraServer Kapasitas dapat digunakan dengan jenis dan AWS wilayah instance berikut.

> **Note**
>
> Ukuran Blok Kapasitas 64 instans tidak didukung untuk semua jenis instans secara keseluruhan Wilayah AWS.

#### Blok Kapasitas Instance

| Tipe instan | US Eas (N. Virg us-eas | US Eas (Oh us-eas | US We (N. Cal a) us-wes | US We (Or us-wes | Eur (Sto m) eu-nor 1 | Eur (Lo eu-wes | Ero (Sp eu-sou 2 | Asi Pac (To ap-nor ast- | Asi Pac (Se ap-nor ast- | Asi Pac (Mu ap-sou 1 | Asi Pas (Hy d) ap-sou 2 | Asi Pac (Sy ap-sou ast- | Asi Pas (Jal ap-Ter ra 3 | Asi Pas (Me e) ap-Ter ra -4 | Am Sel (Sa Pau sa-eas | AW Gov (US Eas us-gov ea st-1 | AW Gov (US We us-gov we st-1 | AS Tim (Atl us-timu 1-atl-2 | AS Barat (Phoenix) us-barat-2-phx-2a |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| p6-b300.8xlar | ✓ | | | ✓ | | | | | ✓ | | ✓ | | ✓ | | | ✓ | | ✓ | |

| Tipe instan | US Eas (N. Virg us-eas | US Eas (Oh us-eas | US We (N. Cal a) us-wes | US We (Or us-wes | Eur (Sto m) eu-nor 1 | Eur (Lo eu-wes | Ero (Sp eu-sou 2 | Asi Pac (To ap-nor ast- | Asi Pac (Se ap-nor ast- | Asi Pac (Mu ap-sou 1 | Asi Pas (Hyi d) ap-sou 2 | Asi Pac (Sy ap-sou ast- | Asi Pas (Jal ap-Ter ra 3 | Asi Pas (Me e) ap-Ter ra -4 | Am Sel (Sa Pau sa-eas | AW Gov (US Eas us-gov ea st-1 | AW Gov (US We us-gov we st-1 | AS Tim (Atl us-timu 1-atl-2 | AS Barat (Phoenix) us-barat-2-phx-2a |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| p6-b200. 8xla | ✓ | ✓ | | ✓ | | | | | | ✓ | ✓ | | | | | ✓ | ✓ | | |
| p5.4x e | ✓ | ✓ | | ✓ | | ✓ | | ✓ | | ✓ | | ✓ | | | ✓ | | | | |
| p5.48 ge | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | | ✓ | | ✓ | ✓ | | ✓ | | | ✓ | |
| p5e.4 rge | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | | ✓ | | ✓ | ✓ | | ✓ | | | | ✓ |
| p5en. arge | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | ✓ | | | | | | |
| p4d.2 rge | ✓ | ✓ | | ✓ | | | | | | | | | | | | | | | |
| p4de. arge | ✓ | | | ✓ | | | | | | | | | | | | | | | |
| trn1. arge | ✓ | ✓ | | ✓ | ✓ | | | | | ✓ | | ✓ | | ✓ | | | | | |
| trn2. rge | | | | | | | | | | | | | | ✓ | ✓ | | | | |
| trn2. arge | | ✓ | | | | | | | | | | | | | | | | | |

UltraServer Blok Kapasitas

| Tipe instans | US East (Ohio) us-east-2 | AS Timur (Dallas) us-timur-1-dfw-2a |
|---|---|---|
| `Trn2` | ✓ | |
| `P6e-GB200` | | ✓ |

Platform yang didukung

Blok Kapasitas untuk ML saat ini mendukung instance dan hanya UltraServers dengan sewa default. Saat Anda menggunakan Konsol Manajemen AWS untuk membeli Blok Kapasitas, opsi platform default adalah Linux/UNIX. Saat Anda menggunakan AWS Command Line Interface (AWS CLI) atau AWS SDK membeli Blok Kapasitas, opsi platform berikut tersedia:

- Linux/Unix
- Linux Red Hat Enterprise
- RHEL dengan HA
- SUSE Linux
- Ubuntu Pro

Pertimbangan-pertimbangan

Sebelum Anda menggunakan Blok Kapasitas, pertimbangkan detail dan batasan berikut.

- Jika penurunan terdeteksi pada Blok UltraServer Kapasitas, Anda akan diberi tahu tetapi umumnya instans Anda tidak dihentikan pada Blok Kapasitas. Ini untuk meminimalkan gangguan yang tidak diinginkan pada beban kerja Anda. Anda dapat terus menggunakan Blok UltraServer Kapasitas apa adanya setelah menerima pemberitahuan ini atau meminta perbaikan dengan mengakhiri semua instans pada blok kapasitas dan mengirimkan kasus AWS dukungan. Setelah kasus dukungan diterima, Anda akan diberi tahu saat remediasi selesai dan Anda dapat meluncurkan kembali instans ke Blok Kapasitas Anda UltraServer .
- Untuk Blok `P6e-GB200` UltraServer Kapasitas, Anda harus menghentikan instans setidaknya 60 menit sebelum waktu akhir Blok Kapasitas.

- Untuk membeli dan menggunakan Blok Kapasitas di Zona Lokal, Anda harus memilih untuk masuk ke Zona Lokal.
- Setiap Blok Kapasitas dapat memiliki hingga 64 instans, dan Anda dapat memiliki hingga 256 instans di seluruh Blok Kapasitas.
- Anda dapat menjelaskan penawaran Blok Kapasitas yang dapat dimulai segera setelah 30 menit.
- Blok Kapasitas berakhir pada pukul 11:30 Waktu Universal Terkoordinasi (UTC).
- Proses pengakhiran untuk instans yang berjalan di Blok Kapasitas dimulai pada pukul 11:00 Waktu Universal Terkoordinasi (UTC) pada hari terakhir reservasi.
- Blok Kapasitas dapat dipesan dengan waktu mulai hingga 8 minggu di masa mendatang.
- Pembatalan Blok Kapasitas tidak diperbolehkan.
- UltraServer Blok Kapasitas tidak dapat dibagikan di seluruh AWS akun atau di dalam AWS Organisasi Anda.
- Blok Kapasitas tidak dapat dipindahkan atau dibagi.
- Anda dapat menambahkan Blok UltraServer Kapasitas dan Blok Kapasitas instans ke Grup Sumber Daya Reservasi Kapasitas. Untuk informasi selengkapnya, lihat Grup Sumber Daya Reservasi Kapasitas.
- Blok Kapasitas tidak mendukung grup penempatan. Anda tidak dapat menentukan grup penempatan saat meluncurkan instans ke Blok Kapasitas, apakah Anda menargetkan Blok Kapasitas berdasarkan ID reservasi atau melalui Grup Sumber Daya Reservasi Kapasitas.
- Jumlah total instans yang dapat dipesan di Blok Kapasitas di semua akun di AWS Organisasi Anda tidak boleh melebihi 256 instans pada tanggal tertentu.
- Untuk menggunakan Blok Kapasitas, instans harus secara khusus menargetkan ID reservasi.
- Instans dalam Blok Kapasitas tidak dihitung terhadap batas On-Demand Instans Anda.
- Untuk instans P5 menggunakan AMI khusus, pastikan Anda memiliki perangkat lunak dan konfigurasi yang diperlukan untuk EF A.
- Untuk grup node terkelola Amazon EKS, lihat Membuat grup node terkelola dengan Blok Kapasitas Amazon EC2 untuk ML. Untuk grup node yang dikelola sendiri Amazon EKS, lihat Menggunakan Blok Kapasitas untuk ML dengan node yang dikelola sendiri.

Sumber daya terkait

Setelah Anda membuat Blok Kapasitas, Anda dapat melakukan hal berikut dengan Blok Kapasitas:

- Luncurkan instance ke Blok Kapasitas. Untuk informasi selengkapnya, lihat Luncurkan instans menggunakan Blok Kapasitas.
- Buat grup Penskalaan Otomatis Amazon EC2. Untuk informasi selengkapnya, lihat Menggunakan Blok Kapasitas untuk beban kerja pembelajaran mesin di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

> **Note**
>
> Jika Anda menggunakan Amazon EC2 Auto Scaling atau Amazon EKS, Anda dapat menjadwalkan penskalaan untuk dijalankan pada awal reservasi Blok Kapasitas. Dengan penskalaan terjadwal, AWS secara otomatis menangani percobaan ulang untuk Anda, jadi Anda tidak perlu khawatir menerapkan logika coba ulang untuk menangani kegagalan sementara.

- Tingkatkan alur kerja ML dengan Layanan Komputasi AWS Paralel. Untuk informasi selengkapnya, lihat Dukungan Blo k Kapasitas untuk Layanan Komputasi AWS Paralel.
- Tingkatkan alur kerja ML dengan AWS ParallelCluster. Untuk informasi selengkapnya, lihat Meningkatkan alur kerja ML dengan AWS ParallelCluster dan Blok Kapasitas Amazon EC2 untuk ML.

Untuk informasi selengkapnya tentang Layanan AWS Komputasi Paralel, lihat Apa itu Layanan Komputasi AWS Paralel.

Untuk informasi selengkapnya AWS ParallelCluster, lihat Apa itu AWS ParallelCluster.

Cara kerja Blok Kapasitas Amazon EC2

Anda dapat memesan Blok Kapasitas dengan spesifikasi sebagai berikut:

- Pesan waktu mulai hingga 8 minggu sebelumnya
- Tetapkan durasi reservasi satu hingga 14 hari atau kelipatan 7 hari, hingga 182 hari (Contoh: 21 hari, 28 hari)
- Konfigurasikan hingga 64 instans per Blok Kapasitas
- Konfigurasikan hingga 256 instans di beberapa Blok Kapasitas

Untuk Amazon EC2 UltraServers, masing-masing UltraServer sesuai dengan satu Blok Kapasitas. Anda dapat meminta beberapa UltraServers melalui satu permintaan.

Untuk memesan Blok Kapasitas, Anda mulai dengan menentukan kebutuhan kapasitas Anda, termasuk jenis atau UltraServer jenis instans, jumlah instans atau UltraServers, jumlah waktu, tanggal mulai paling awal, dan tanggal akhir terbaru yang Anda butuhkan. Kemudian, Anda dapat melihat penawaran Blok Kapasitas yang tersedia yang memenuhi spesifikasi Anda. Penawaran Blok Kapasitas mencakup detail seperti waktu mulai, Zona Ketersediaan, dan harga reservasi. Harga penawaran Blok Kapasitas tergantung pada penawaran dan permintaan yang tersedia pada saat penawaran dikirimkan. Setelah Anda memesan Blok Kapasitas, harga tidak berubah. Untuk informasi selengkapnya, lihat Harga dan penagihan Blok Kapasitas.

Saat Anda membeli penawaran Blok Kapasitas, reservasi dibuat sesuai tanggal dan jumlah instans yang Anda pilih. Saat reservasi Blok Kapasitas dimulai, Anda dapat menargetkan peluncuran instans dengan menentukan ID reservasi dalam permintaan peluncuran.

Anda dapat menggunakan semua instans yang Anda pesan hingga 30 menit (misalnya jenis) atau 60 menit (untuk UltraServer jenis) sebelum waktu akhir Blok Kapasitas. Dengan 30 menit (misalnya jenis) atau 60 menit (untuk UltraServer jenis) tersisa dalam reservasi Blok Kapasitas Anda, instans apa pun yang berjalan di Blok Kapasitas mulai dihentikan. Waktu ini digunakan untuk membersihkan instans Anda sebelum mengirimkan Blok Kapasitas ke pelanggan berikutnya. Suatu peristiwa dipancarkan melalui EventBridge 10 menit sebelum proses penghentian dimulai. Untuk informasi selengkapnya, lihat Memantau Blok Kapasitas menggunakan EventBridge.

### Harga dan penagihan Blok Kapasitas

Dengan Blok Kapasitas Amazon EC2 untuk M, Anda hanya membayar untuk apa yang Anda pesan. Harga Blok Kapasitas tergantung pada penawaran dan permintaan yang tersedia untuk Blok Kapasitas pada saat pembelian. Anda dapat melihat harga penawaran Blok Kapasitas sebelum Anda memesannya. Harga Blok Kapasitas dibebankan di muka pada saat reservasi dilakukan. Saat Anda mencari Blok Kapasitas di berbagai tanggal, penawaran Blok Kapasitas dengan harga terendah yang tersedia akan dikembalikan. Setelah Anda memesan Blok Kapasitas, harga tidak berubah.

Ketika Anda menggunakan Blok Kapasitas, Anda membayar untuk sistem operasi yang Anda gunakan saat instans Anda berjalan. Untuk informasi selengkapnya tentang harga sistem operasi, lihat Blok Kapasitas Amazon EC2 untuk Harga ML.

#### Penagihan

Harga penawaran Blok Kapasitas dibebankan di muka. Pembayaran ditagih ke AWS akun Anda dalam waktu 5 menit hingga 12 jam setelah Anda membeli Blok Kapasitas. Saat pembayaran Anda diproses, sumber daya reservasi Blok Kapasitas Anda tetap dalam status `payment-pending`. Jika

pembayaran Anda tidak dapat diproses setidaknya 5 menit sebelum waktu mulai pemblokiran, atau dalam waktu 12 jam (mana yang lebih dulu), Blok Kapasitas Anda dilepaskan dan status reservasi berubah menjadi`payment-failed`.

Setelah pembayaran Anda berhasil diproses, status sumber daya Blok Kapasitas berubah dari `payment-pending` menjadi `scheduled`. Anda menerima faktur yang menunjukkan pembayaran satu kali di muka. Dalam faktur, Anda dapat mengaitkan jumlah yang dibayarkan dengan ID reservasi Blok Kapasitas.

Ketika reservasi Blok Kapasitas dimulai, Anda ditagih hanya berdasarkan sistem operasi yang Anda gunakan saat instans Anda berjalan di reservasi. Anda dapat melihat penggunaan dan biaya terkait dalam tagihan setahun Anda untuk bulan penggunaan di AWS Cost and Usage Report Anda.

> **Note**
>
> Diskon Savings Plans dan Reserved instans tidak berlaku untuk Blok Kapasitas.

### Melihat tagihan Anda

Anda dapat melihat tagihan Anda di AWS Manajemen Penagihan dan Biaya konsol. Pembayaran di muka untuk Blok Kapasitas Anda muncul di bulan pembelian reservasi.

Setelah reservasi dimulai, tagihan Anda menunjukkan reservasi blok yang digunakan dan waktu yang tidak digunakan dalam baris yang terpisah. Anda dapat menggunakan item baris ini untuk melihat berapa banyak waktu yang digunakan dalam reservasi Anda. Anda hanya akan melihat biaya penggunaan di baris untuk waktu yang digunakan jika Anda menggunakan sistem operasi premium. Untuk informasi selengkapnya, lihat Harga dan penagihan Blok Kapasitas. Tidak ada biaya tambahan untuk waktu yang tidak digunakan.

Untuk informasi selengkapnya, lihat Melihat tagihan Anda dalam Panduan Pengguna AWS Manajemen Penagihan dan Biaya .

Jika Blok Kapasitas dimulai pada bulan yang berbeda dari bulan pembelian reservasi, harga di muka dan penggunaan reservasi muncul di bawah bulan tagihan yang terpisah. Di Anda AWS Cost and Usage Report, ID reservasi Blok Kapasitas tercantum dalam item  reservation/ReservationARN baris biaya dimuka dan  lineitem/ResourceID  di tagihan ulang tahun Anda sehingga Anda dapat mengaitkan penggunaan dengan harga di muka yang sesuai.

### Temukan dan beli Blok Kapasitas

Untuk memesan Blok Kapasitas, pertama-tama Anda harus menemukan blok waktu ketika kapasitas tersedia yang sesuai dengan kebutuhan Anda. Untuk menemukan Blok Kapasitas yang tersedia untuk dipesan, Anda menentukan yang berikut ini.

- Jumlah instans yang Anda butuhkan
- Durasi waktu Anda membutuhkan instans
- Rentang tanggal yang Anda perlukan untuk reservasi

Untuk mencari penawaran Blok Kapasitas yang tersedia, tentukan durasi reservasi dan jumlah instans. Anda harus menentukan durasi reservasi dengan  penambahan 1 hari hingga 14 hari, dan dalam penambahan 7 hari hingga 182 hari.  Setiap Blok Kapasitas dapat memiliki hingga 64 instans, dan Anda dapat memiliki hingga 256 instans di seluruh Blok Kapasitas.

Saat Anda meminta Blok Kapasitas yang sesuai dengan spesifikasi Anda, detail hingga 6 blok yang tersedia dikembalikan. Semua Blok Kapasitas berakhir pada pukul 11:30 WIB, sehingga blok yang dimulai pada hari yang sama akan memiliki durasi yang paling cocok dengan durasi yang Anda inginkan. Satu blok akan memiliki durasi yang sedikit kurang dari durasi yang Anda inginkan, sementara yang lain akan memiliki durasi sedikit lebih besar dari durasi yang Anda inginkan.

Detail penawaran termasuk waktu mulai reservasi, Zona Ketersediaan untuk reservasi, dan harga reservasi. Untuk informasi selengkapnya, lihat Harga dan penagihan Blok Kapasitas.

Anda dapat membeli penawaran Blok Kapasitas yang ditampilkan, atau Anda dapat memodifikasi kriteria pencarian untuk melihat opsi lain yang tersedia. Tidak ada waktu kedaluwarsa yang telah ditentukan untuk penawaran, tetapi penawaran hanya tersedia berdasarkan siapa cepat, dia dapat.

Ketika Anda membeli penawaran Blok Kapasitas, Anda mendapatkan tanggapan langsung yang mengonfirmasi bahwa Blok Kapasitas Anda telah terpesan. Setelah konfirmasi, Anda akan melihat Reservasi Kapasitas baru di akun Anda dengan tipe reservasi `capacity-block` dan `start-date` diatur ke waktu mulai penawaran yang Anda beli. Reservasi Blok Kapasitas Anda dibuat dengan status `payment-pending`. Setelah pembayaran di muka berhasil diproses, status reservasi berubah menjadi `scheduled`. Untuk informasi selengkapnya, lihat Penagihan.

> **Note**
>
> Untuk membeli dan menggunakan Blok Kapasitas di Zona Lokal, Anda harus memilih untuk masuk ke Zona Lokal.

Console

Untuk menemukan dan membeli Blok Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di bilah navigasi di bagian atas layar, pilih Wilayah AWS. Pilihan ini penting karena ukuran Blok Kapasitas 64 instans tidak didukung untuk semua jenis instans di semua Wilayah.
3. Di panel navigasi, pilih Reservasi Kapasitas, Buat Blok Kapasitas.
4. Di bawah Jenis Blok Kapasitas, pilih Instans atau UltraServers.
5. Di bawah Atribut kapasitas, Anda dapat menentukan parameter pencarian Blok Kapasitas. Secara default, platformnya adalah Linux. Jika Anda ingin memilih sistem operasi yang berbeda, gunakan AWS CLI. Untuk informasi selengkapnya, lihat Platform yang didukung.
6. Di bawah Kapasitas total (untuk Instans) atau UltraServer hitungan (untuk UltraServers), tentukan jumlah instans atau UltraServers yang ingin Anda pesan.
7. Di bawah Durasi, masukkan jumlah hari atau minggu yang Anda butuhkan untuk reservasi.
8. Di bawah Rentang tanggal untuk mencari Blo k Kapasitas, masukkan tanggal paling awal yang ingin reservasi dimulai.
9. Pilih Temukan Blok Kapasitas.
10. Jika Blok Kapasitas tersedia yang memenuhi spesifikasi Anda, Anda akan melihat penawaran di bawah Blok Kapasitas yang Disarankan. Jika ada beberapa penawaran yang memenuhi spesifikasi Anda, penawaran Blok Kapasitas paling awal yang tersedia akan ditampilkan. Untuk melihat penawaran Blok Kapasitas lainnya, sesuaikan input pencarian Anda dan pilih Temukan Blok Kapasitas lagi.
11. Ketika Anda menemukan penawaran Blok Kapasitas yang ingin Anda beli, pilih Berikutnya.
12. (Opsional) Pada halaman Tambahkan tanda, pilih Tambahkan tanda baru.
13. Halaman Tinjauan dan pembelian menampilkan daftar tanggal mulai dan berakhir, durasi, jumlah total instans, dan harga.

> **Note**
>
> Blok Kapasitas tidak dapat dibatalkan setelah Anda memesannya.

14. Di jendela popup Beli Blok Kapasitas, ketik konfirmasi, lalu pilih Beli.

AWS CLI

Untuk menemukan Blok Kapasitas instans

Gunakan perintah describe-capacity-block-offers.

Contoh berikut menemukan blok kapasitas instance.

```
aws ec2 describe-capacity-block-offerings \
--instance-type p5.48xlarge \
--instance-count 16 \
--start-date-range 2023-08-14T00:00:00Z \
--end-date-range 2023-10-22-T00:00:00Z \
--capacity-duration-hours 48 \
--all-availability-zones
```

Contoh berikut menemukan Blok UltraServer Kapasitas.

```
aws ec2 describe-capacity-block-offerings \
--ultraserver-type u-p6e-gb200x72 \
--ultraserver-count 1 \
--start-date-range 2023-08-14T00:00:00Z \
--end-date-range 2023-10-22-T00:00:00Z \
--capacity-duration-hours 48
```

Untuk membeli Blok Kapasitas

Gunakan perintah purchase-capacity-block dengan ID penawaran Blok Kapasitas dari output contoh sebelumnya.

```
aws ec2 purchase-capacity-block \
--capacity-block-offering-id cb-0123456789abcdefg \
--instance-platform Linux/UNIX
```

PowerShell

Untuk menemukan Blok Kapasitas

Gunakan Get-EC2CapacityBlockOffering cmdlet.

Contoh berikut menemukan blok kapasitas instance.

```
Get-EC2CapacityBlockOffering `
-InstanceType p5.48xlarge `
-InstanceCount 16 `
-CapacityDurationHour 48 `
-StartDateRange 2023-08-14T00:00:00Z `
-EndDateRange 2023-10-22-T00:00:00Z `
-AllAvailabilityZones $true
```

Contoh berikut menemukan Blok UltraServer Kapasitas.

```
Get-EC2CapacityBlockOffering `
-UltraserverType u-p6e-gb200x72 `
-UltraserverCount  1 `
-CapacityDurationHour 48 `
-StartDateRange 2023-08-14T00:00:00Z `
-EndDateRange 2023-10-22-T00:00:00Z
```

Untuk membeli Blok Kapasitas

Gunakan New-EC2EC2CapacityBlock cmdlet dengan ID penawaran Blok Kapasitas dari output contoh sebelumnya.

```
New-EC2EC2CapacityBlock `
-CapacityBlockOfferingId cb-0123456789abcdefg `
-InstancePlatform Linux/UNIX
```

Luncurkan instans menggunakan Blok Kapasitas

Untuk menggunakan Blok Kapasitas, Anda harus menentukan ID reservasi Blok Kapasitas saat meluncurkan instans. Meluncurkan sebuah instans ke dalam Blok Kapasitas mengurangi kapasitasnya yang tersedia dengan jumlah instans yang diluncurkan. Misalnya, jika kapasitas instans

yang Anda beli adalah delapan instans dan Anda meluncurkan empat instans, kapasitas yang tersedia dikurangi empat instans.

Jika Anda mengakhiri instans yang berjalan di Blok Kapasitas sebelum reservasi berakhir, Anda dapat meluncurkan instans baru sebagai gantinya. Saat Anda mengakhiri atau menghentikan instans di Blok Kapasitas, dibutuhkan waktu beberapa menit untuk membersihkan instans sebelum Anda dapat meluncurkan instans lain untuk menggantinya. Selama waktu ini, instans Anda akan dalam status berhenti atau `shutting-down`. Setelah proses ini selesai, status instans Anda akan berubah menjadi `stopped` atau `terminated`. Kemudian, kapasitas yang tersedia di Blok Kapasitas Anda akan diperbarui untuk menampilkan instans lain yang tersedia untuk digunakan.

Persyaratan

- Instans Anda tidak dapat diluncurkan di subnet di Zona Ketersediaan yang berbeda dari Zona Ketersediaan tempat Blok Kapasitas Anda berada.
- Instans Anda tidak dapat diluncurkan menggunakan AMI dengan platform yang berbeda dari platform untuk Blok Kapasitas Anda.
- Untuk menggunakan Blok Kapasitas di Zona Lokal, Anda harus memilih untuk masuk ke Zona Lokal.

Console

Untuk meluncurkan instans ke Blok Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada bilah navigasi di bagian atas layar, pilih Wilayah untuk reservasi Blok Kapasitas Anda.
3. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.
4. Ikuti prosedur untuk meluncurkan instans.
5. Perluas Detail lanjutan, dan untuk opsi Pembelian, pilih Blo k Kapasitas. Kemudian lakukan salah satu hal berikut:
   - Untuk meluncurkan instans ke Blok Kapasitas tertentu, untuk Reservasi Kapasitas pilih Tentukan Reservasi Kapasitas, lalu pilih Blok Kapasitas.
   - (UltraServers hanya) Untuk meluncurkan instans ke Grup Sumber Daya Reservasi UltraServer Kapasitas Blok Kapasitas, untuk Reservasi Kapasitas pilih T entukan Grup Sumber Daya Reservasi Kapasitas, lalu pilih Grup Sumber Daya Reservasi Kapasitas.
6. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan instans menggunakan ke Blok Kapasitas

Gunakan perintah run-instance dengan opsi. `instance-market-options MarketType`

Contoh berikut meluncurkan instance ke Blok Kapasitas tertentu.

```
aws ec2 run-instances \
--image-id ami-0abcdef1234567890 \
--count 1 \
--instance-type p5.48xlarge \
--key-name my-key-pair \
--subnet-id subnet-0abcdef1234567890 \
--instance-market-options MarketType='capacity-block' \
--capacity-reservation-specification
 CapacityReservationTarget={CapacityReservationId=capacity_block_id}
```

Contoh berikut meluncurkan instance ke Grup Sumber Daya Reservasi UltraServer Kapasitas Blok Kapasitas.

```
aws ec2 run-instances \
--image-id ami-0abcdef1234567890 \
--count 1 \
--instance-type p6e-gb200.36xlarge \
--key-name my-key-pair \
--subnet-id subnet-0abcdef1234567890 \
--instance-market-options MarketType='capacity-block' \
--capacity-reservation-specification
 CapacityReservationTarget={CapacityReservationResourceGroupArn=resource_group_arn}
```

PowerShell

Untuk meluncurkan instans ke Blok Kapasitas

Gunakan New-EC2Instance cmdlet dengan `-InstanceMarketOption` opsi yang didefinisikan sebagai berikut.

```
$marketoption = New-Object Amazon.EC2.Model.InstanceMarketOptionsRequest
$marketoption.MarketType = "capacity-block"
```

Contoh berikut meluncurkan instance ke Blok Kapasitas tertentu.

```
New-EC2Instance `
-ImageId ami-0abcdef1234567890 `
-InstanceType p5.48xlarge `
-KeyName "my-key-pair" `
-SubnetId subnet-0abcdef1234567890 `
-InstanceMarketOptions $marketoption `
-CapacityReservationTarget_CapacityReservationId capacity_block_id
```

Contoh berikut meluncurkan instance ke Grup Sumber Daya Reservasi UltraServer Kapasitas Blok Kapasitas.

```
New-EC2Instance `
-ImageId ami-0abcdef1234567890 `
-InstanceType p6e-gb200.36xlarge `
-KeyName "my-key-pair" `
-SubnetId subnet-0abcdef1234567890 `
-InstanceMarketOptions $marketoption `
-CapacityReservationTarget_CapacityReservationResourceGroupArn "resource_group_arn"
```

### Sumber daya terkait

- Untuk membuat template peluncuran yang menargetkan Blok Kapasitas, lihatthe section called “Templat peluncuran”.
- Untuk meluncurkan instans ke Blok Kapasitas menggunakan EC2 Fleet, lihatTutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Blok Kapasitas.
- Untuk mengatur grup node terkelola EKS dengan Blok Kapasitas, lihat Membuat grup node terkelola dengan Blok Kapasitas untuk ML di Panduan Pengguna Amazon EKS .
- Untuk mengatur AWS ParallelCluster menggunakan Blok Kapasitas, lihat ML aktif AWS ParallelCluster.

### Melihat Blok Kapasitas

Setelah Anda memesan Blok Kapasitas, Anda dapat melihat reservasi Blok Kapasitas di akun AWS Anda. Anda dapat menampilkan `start-date` dan `end-date` untuk melihat kapan reservasi Anda akan dimulai dan berakhir. Sebelum reservasi Blok Kapasitas dimulai, kapasitas yang tersedia muncul adalah nol. Anda dapat melihat berapa banyak instans yang akan tersedia di Blok Kapasitas Anda dengan nilai tanda untuk kunci tanda `aws:ec2capacityreservation:incrementalRequestedQuantity`.

Saat reservasi Blok Kapasitas dimulai, status reservasi berubah dari `scheduled` menjadi `active`. Sebuah acara dipancarkan melalui Amazon EventBridge untuk memberi tahu Anda bahwa Blok Kapasitas tersedia untuk digunakan. Untuk informasi selengkapnya, lihat Memantau Blok Kapasitas menggunakan EventBridge.

Blok Kapasitas memiliki status sebagai berikut:

- `payment-pending` – Pembayaran dimuka belum diproses.
- `payment-failed`—Pembayaran tidak dapat diproses dalam jangka waktu 12 jam. Blok Kapasitas Anda telah dirilis.
- `scheduled` – Pembayaran telah diproses dan reservasi Blok Kapasitas belum dimulai.
- `active` – Kapasitas terpesan tersedia untuk Anda gunakan.
- `expired` – Reservasi Blok Kapasitas kedaluwarsa secara otomatis pada tanggal dan waktu yang ditentukan dalam permintaan reservasi Anda. Kapasitas terpesan tidak lagi tersedia untuk Anda gunakan.

Console

Untuk melihat Blok Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pada halaman ikhtisar Reservasi Kapasitas, Anda melihat tabel sumber daya dengan detail tentang semua sumber daya Reservasi Kapasitas Anda. Untuk menemukan reservasi Blok Kapasitas, pilih Blok Kapasitas dari daftar tarik-turun di atas ID Reservasi Kapasitas. Dalam tabel, Anda dapat melihat informasi tentang Blok Kapasitas seperti tanggal mulai dan berakhir, durasi, dan status.
4. Untuk detail selengkapnya tentang Blok Kapasitas, pilih ID reservasi untuk Blok Kapasitas yang ingin Anda lihat. Halaman detail Reservasi Kapasitas menampilkan semua properti reservasi dan jumlah instans yang digunakan serta tersedia di Blok Kapasitas.

> **Note**
>
> Sebelum reservasi Blok Kapasitas dimulai, kapasitas yang tersedia muncul adalah nol. Anda dapat melihat berapa banyak instans yang akan tersedia saat reservasi

Blok Kapasitas dimulai dengan menggunakan nilai tanda berikut untuk kunci tanda: `aws:ec2capacityreservation:incrementalRequestedQuantity`.

AWS CLI

Untuk melihat Blok Kapasitas

Gunakan perintah describe-capacity-blocks untuk melihat detail tentang reservasi Blok Kapasitas Anda.

```
aws ec2 describe-capacity-blocks
```

PowerShell

Untuk melihat Blok Kapasitas

Gunakan Get-EC2CapacityBlock cmdlet untuk melihat detail tentang reservasi Blok Kapasitas Anda.

```
Get-EC2CapacityBlock
```

Perluas Blok Kapasitas

Dengan Blok Kapasitas, Anda memesan kapasitas komputasi untuk beban kerja Anda, memastikan ketersediaan dan konsistensi. Untuk mengakomodasi perubahan kebutuhan Anda, Anda dapat memperpanjang durasi Blok Kapasitas yang ada sesuai kebutuhan.

Untuk memperpanjang Blok Kapasitas, itu harus memiliki status `active` atau`scheduled`, dan tidak memiliki ekstensi yang ada`payment-pending`. Anda dapat meminta untuk memperpanjang durasi Blok Kapasitas Anda hingga minimal 1 jam atau maksimum 57 hari sebelum kedaluwarsa. Anda dapat memperpanjang Blok Kapasitas Anda dengan penambahan 1 hari hingga 14 hari, dan penambahan 7 hari hingga total 182 hari (26 minggu). Saat Anda memperpanjang Blok Kapasitas, tanggal berakhirnya akan diperbarui sehingga instans Anda dapat terus berjalan tanpa gangguan.

- Tidak ada batasan jumlah ekstensi yang dapat Anda terapkan ke Blok Kapasitas
- ID Reservasi Kapasitas Anda akan tetap sama setelah memperpanjang blok
- Blok Kapasitas hanya dapat diperpanjang jika ada kapasitas yang cukup tersedia untuk mendukungnya, yang tidak dijamin.

Penagihan

Harga penawaran Blok Kapasitas dibebankan di muka. Perpanjangan akan tetap `payment-pending` berlaku sampai tagihan dibayar. Jika pembayaran Anda tidak dapat diproses dalam waktu 12 jam, atau hingga 35 menit sebelum Blok Kapasitas dijadwalkan berakhir (mana yang lebih dulu), ekstensi Anda tidak berhasil dan statusnya berubah menjadi`payment-failed`. Reservasi Blok Kapasitas Anda akan tetap ada `active` dan akan dihentikan pada tanggal akhir semula.

Setelah pembayaran berhasil diproses, status ekstensi Blok Kapasitas berubah menjadi `payment-succeeded` dan tanggal akhir reservasi Blok Kapasitas akan diperbarui ke tanggal akhir yang baru. Detail ekstensi Anda dapat dilihat di bagian detail Ekstensi Blok Kapasitas di konsol, atau dengan menggunakan perintah describe-capac ity-block-extension-history.

Perluas Blok Kapasitas Anda

Console

Untuk memperpanjang Blok Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pada halaman ik htisar Reservasi Kapasitas, Anda melihat tabel sumber daya dengan detail tentang semua sumber daya Reservasi Kapasitas Anda. Pilih ID reservasi untuk Blok Kapasitas yang ingin Anda perpanjang.
4. Dari menu tar ik -turun Actions, pilih Ex tend Capacity Block.
5. Di bawah Durasi, masukkan jumlah hari atau minggu yang Anda perlukan untuk memperpanjang reservasi.
6. Pilih Temukan Blok Kapasitas.
7. Jika Blok Kapasitas tersedia yang memenuhi spesifikasi Anda, penawaran akan muncul di bawah Blok Kapasitas yang Di rekomendasikan. Untuk melihat penawaran Blok Kapasitas lainnya, sesuaikan input pencarian Anda dan pilih Temukan Blok Kapasitas lagi.
8. Ketika Anda menemukan penawaran Blok Kapasitas yang ingin Anda beli, pilih Per panjang.
9. Di jendela pop-up Per panjang Blok Kapasitas, masukkan konfirmasi, lalu pilih Per panjang.

AWS CLI

Untuk menemukan ekstensi Blok Kapasitas

Gunakan perintah describe-capacity-block-extension-offers. Contoh berikut mencari ekstensi Blok Kapasitas 48 jam untuk Reservasi yang ditentukan.

```
aws ec2 describe-capacity-block-extension-offerings \
    --capacity-reservation-id cr-1234567890abcdefg \
    --capacity-block-extension-duration-hours 48
```

Untuk memperpanjang Blok Kapasitas

Gunakan perintah purchase-capacity-block-extension. Tentukan ID penawaran ekstensi dari output contoh sebelumnya.

```
aws ec2 purchase-capacity-block-extension \
    --capacity-block-extension-offering-id cbe-0123456789abcdefg \
    --capacity-reservation-id cr-1234567890abcdefg
```

PowerShell

Untuk menemukan ekstensi Blok Kapasitas

Gunakan Get-EC2CapacityBlockExtensionOffering cmdlet. Contoh berikut mencari ekstensi Blok Kapasitas 48 jam untuk Reservasi yang ditentukan.

```
Get-EC2CapacityBlockExtensionOffering `
    -CapacityReservationId cr-1234567890abcdefg `
    -CapacityBlockExtensionDurationHour 48
```

Untuk memperpanjang Blok Kapasitas

Gunakan Invoke-EC2CapacityBlockExtension cmdlet. Tentukan ID penawaran ekstensi dari output contoh sebelumnya.

```
Invoke-EC2CapacityBlockExtension `
    -CapacityBlockExtensionOfferingId cbe-0123456789abcdefg `
    -CapacityReservationId cr-1234567890abcdefg
```

## Bagikan Blok Kapasitas

Berbagi Blok Kapasitas memungkinkan pemilik Blok Kapasitas berbagi Blok Kapasitas Amazon EC2 dengan AWS akun lain dalam AWS Organisasi. Ini memungkinkan Anda memaksimalkan

pemanfaatan kapasitas GPU cadangan di berbagai tim dan proyek untuk menggunakan Blok Kapasitas secara efisien.

AWS Akun yang memiliki Blok Kapasitas (pemilik) dapat membagikannya dengan AWS akun lain (konsumen). Pemilik dapat berbagi Blok Kapasitas dengan AWS akun tertentu di dalam AWS Organisasi mereka, unit organisasi di dalam AWS Organisasi mereka, atau seluruh AWS Organisasi. Konsumen dapat meluncurkan instans ke Blok Kapasitas yang dibagikan dengan mereka dengan cara yang sama seperti mereka meluncurkan instans ke Blok Kapasitas yang mereka miliki.

### Prasyarat untuk berbagi Blok Kapasitas

Sebelum Anda dapat berbagi Blok Kapasitas, kondisi berikut harus dipenuhi:

- Anda harus memiliki Blok Kapasitas - Anda tidak dapat berbagi Blok Kapasitas yang telah dibagikan dengan Anda.
- Status Blok Kapasitas harus aktif atau terjadwal - Blok Kapasitas yang berada di status lain, seperti `expired` atau `payment-pending` tidak dapat dibagikan.
- Berbagi hanya dalam AWS Organisasi Anda - Pemilik dapat berbagi Blok Kapasitas dengan AWS akun tertentu di dalam AWS Organisasi mereka, unit organisasi di dalam AWS Organisasi mereka, atau seluruh AWS Organisasi.
- UltraServer Blok Kapasitas tidak didukung - Anda tidak dapat membagikan Blok Kapasitas untuk Amazon EC2 UltraServers.
- Kelayakan akun - Berbagi Blok Kapasitas tidak tersedia untuk AWS akun baru atau AWS akun yang memiliki riwayat penagihan terbatas.

### Layanan terkait

Berbagi Blok Kapasitas terintegrasi dengan AWS Resource Access Manager (AWS RAM). AWS RAM adalah layanan yang memungkinkan Anda untuk berbagi AWS sumber daya Anda dengan AWS akun apa pun atau melalui AWS Organizations. Dengan AWS RAM, Anda dapat berbagi sumber daya yang Anda miliki dengan membuat berbagi sumber daya. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan dibagikan. Konsumen dapat berupa AWS akun individu, atau unit organisasi atau seluruh organisasi dari AWS Organizations.

Untuk informasi selengkap AWS RAM nya, lihat Panduan AWS RAM Pengguna .

### Izin Blok Kapasitas Bersama

#### Izin untuk pemilik

Pemilik Blok Kapasitas tetap bertanggung jawab untuk mengelola Blok Kapasitas (misalnya, memperluas, berbagi), dan instans yang mereka luncurkan ke dalamnya. Pemilik tidak dapat mengubah instans yang diluncurkan konsumen ke Blok Kapasitas yang telah mereka bagikan.

#### Izin untuk konsumen

Konsumen dapat meluncurkan instans ke kapasitas bersama dan bertanggung jawab untuk mengelola instans tersebut. Konsumen tidak dapat melihat atau memodifikasi instans yang dimiliki oleh konsumen lain atau oleh pemilik Blok Kapasitas. Konsumen juga hanya dapat melihat kapasitas total dan kapasitas yang tersedia di Blok Kapasitas bersama.

### Bagikan Blok Kapasitas

Untuk berbagi Blok Kapasitas, Anda harus menambahkannya ke berbagi sumber daya. Berbagi sumber daya adalah AWS RAM sumber daya yang memungkinkan Anda berbagi sumber daya di seluruh sumber daya Akun AWS.

Jika Anda menambahkan Blok Kapasitas ke pembagian sumber daya yang dibagikan dengan seluruh AWS Organisasi, konsumen di organisasi Anda diberikan akses ke Blok Kapasitas bersama.

#### Console

Untuk membagikan Blok Kapasitas yang Anda miliki menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Blok Kapasitas untuk dibagikan dan pilih Tind akan, Bagikan reservasi.
4. Pilih pembagian sumber daya untuk menambahkan Blok Kapasitas dan pilih Bagikan Reservasi Kapasitas.

   Butuh beberapa menit bagi konsumen untuk mendapatkan akses ke Blok Kapasitas bersama.

Untuk menambahkan Blok Kapasitas ke berbagi sumber daya baru

Anda harus terlebih dahulu membuat berbagi sumber daya menggunakan AWS RAM konsol. Untuk informasi selengkapnya, lihat Membuat berbagi sumber daya di Panduan AWS RAM Pengguna.

AWS CLI

Untuk berbagi Blok Kapasitas yang Anda miliki

Gunakan perintah create-resource-share dan associate-resource-share.

```
aws ram create-resource-share \
    --name my-resource-share \
    --resource-arns arn:aws:ec2:us-east-2:123456789012:capacity-
reservation/cr-1234abcd56EXAMPLE
```

```
aws ram associate-resource-share \
    --resource-share-arn arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-
b505-7e2a-420d-6f5d3EXAMPLE \
    --resource-arns arn:aws:ec2:us-east-2:123456789012:capacity-
reservation/cr-1234abcd56EXAMPLE
```

PowerShell

Untuk berbagi Blok Kapasitas yang Anda miliki

Gunakan Connect-RAMResourceShare cmd New-RAMResourceShare let dan.

```
New-RAMResourceShare `
    -Name my-resource-share `
    -ResourceArn "arn:aws:ec2:us-east-2:123456789012:capacity-
reservation/cr-1234abcd56EXAMPLE"
```

```
Connect-RAMResourceShare `
    -ResourceShareArn "arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-
b505-7e2a-420d-6f5d3EXAMPLE" `
    -ResourceArn "arn:aws:ec2:us-east-2:123456789012:capacity-
reservation/cr-1234abcd56EXAMPLE"
```

Blok Kapasitas beroperasi berdasarkan prinsip pertama datang, pertama dilayani untuk semua akun, terlepas dari status kepemilikannya. Saat Anda berbagi Blok Kapasitas, jika konsumen

meluncurkan instans sebelum pemiliknya, instans tersebut menempati kapasitas hingga konsumen mengakhiri instans atau hingga 30 menit sebelum Blok Kapasitas berakhir.

Berhenti berbagi Blok Kapasitas

Anda dapat berhenti berbagi Blok Kapasitas kapan saja hingga 30 menit sebelum tanggal kedaluwarsa pemblokiran.

Apa yang terjadi ketika Anda berhenti berbagi:

- Konsumen tidak dapat lagi meluncurkan instans baru ke Blok Kapasitas yang tidak dibagikan.
- Instans yang sedang berjalan terus berjalan hingga 30 menit sebelum tanggal kedaluwarsa Blok Kapasitas, kecuali diakhiri oleh konsumen.

Console

Untuk menghentikan berbagi Blok Kapasitas yang Anda miliki menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Blok Kapasitas dan pilih tab Berbagi.
4. Tab Berbagi mencantumkan pembagian sumber daya yang telah ditambahkan Blok Kapasitas. Pilih berbagi sumber daya untuk menghapus Blok Kapasitas.
5. Pilih Hapus dari berbagi sumber daya.

AWS CLI

Untuk berhenti berbagi Blok Kapasitas yang Anda miliki

Gunakan perintah disassociate-resource-share.

```
aws ram disassociate-resource-share \
    --resource-share-arn arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE \
    --resource-arns arn:aws:ec2:us-east-2:123456789012:capacity-reservation/cr-1234abcd56EXAMPLE
```

PowerShell

Untuk berhenti berbagi Blok Kapasitas yang Anda miliki

Gunakan Disconnect-RAMResourceShare cmdlet.

```
Disconnect-RAMResourceShare `
    -ResourceShareArn "arn:aws:ram:us-east-2:123456789012:resource-share/7ab63972-b505-7e2a-420d-6f5d3EXAMPLE" `
    -ResourceArn "arn:aws:ec2:us-east-2:123456789012:capacity-reservation/cr-1234abcd56EXAMPLE"
```

Memantau penggunaan Blok Kapasitas bersama

Pemilik Blok Kapasitas dapat memantau akun mana yang menggunakan Blok Kapasitas bersama mereka dan melacak penggunaan instans per akun.

AWS CLI

Untuk memantau penggunaan Blok Kapasitas

Gunakan perintah get-capacity-reservation-usage .

```
aws ec2 get-capacity-reservation-usage \
    --capacity-reservation-id cr-1234abcd56EXAMPLE
```

API ini memungkinkan pemilik untuk:

- Lihat akun mana yang saat ini menggunakan Blok Kapasitas.
- Lihat jumlah instance yang dijalankan setiap akun.

Pemberitahuan penghentian instans

Akun pemilik dan konsumen yang memiliki instans yang berjalan di Blok Kapasitas akan menerima EventBridge acara 40 menit sebelum reservasi Blok Kapasitas berakhir, yang menunjukkan bahwa setiap instans yang berjalan dalam reservasi akan mulai dihentikan dalam 10 menit. Untuk informasi selengkapnya, lihat Memantau Blok Kapasitas menggunakan EventBridge.

Ekstensi Blok Kapasitas

Blok Kapasitas dapat diperpanjang saat dibagikan. Hanya akun pemilik yang dapat memperpanjang Blok Kapasitas bersama.

Ketika Blok Kapasitas diperpanjang, instans yang sedang berjalan yang diluncurkan oleh pemilik atau konsumen secara otomatis mewarisi tanggal kedaluwarsa baru, dan konsumen dapat terus menggunakan kapasitas bersama hingga tanggal kedaluwarsa baru tanpa gangguan instans apa pun.

Harga dan penagihan

Pemilik ditagih untuk Blok Kapasitas yang mereka bagikan dan membayar di muka untuk Blok Kapasitas saat mereka membelinya. Pemilik juga membayar biaya sistem operasi untuk instans yang mereka jalankan di Blok Kapasitas.

Konsumen hanya ditagih untuk biaya sistem operasi untuk instans yang mereka jalankan di Blok Kapasitas bersama. Konsumen tidak dikenakan biaya untuk reservasi Blok Kapasitas itu sendiri.

Membuat Grup Sumber Daya Reservasi Kapas UltraServer itas untuk Blok Kapasitas

Anda dapat menggunakan AWS Resource Groups untuk membuat koleksi logis Blok UltraServer Kapasitas. Setelah membuat Grup Sumber Daya Reservasi Kapasitas, Anda dapat menambahkan Blo UltraServer k Kapasitas yang Anda miliki di akun Anda. Setelah menambahkan Blok UltraServer Kapasitas, Anda dapat menargetkan peluncuran instans ke Grup Sumber Daya Reservasi Kapasitas alih-alih Blok Kapasitas individual. Instans yang menargetkan Grup Sumber Daya Reservasi Kapasitas cocok dengan Blok UltraServer Kapasitas apa pun dalam grup yang memiliki atribut yang cocok dan kapasitas yang tersedia. Jika Grup Sumber Daya Reservasi Kapasitas tidak memiliki Blok UltraServer Kapasitas dengan atribut yang cocok dan kapasitas yang tersedia, peluncuran instans gagal.

Jika Blok UltraServer Kapasitas dihapus dari Grup Sumber Daya Reservasi Kapasitas saat memiliki instans yang sedang berjalan, instans tersebut akan terus berjalan di Blok Kapasitas. Jika Blok UltraServer Kapasitas dalam grup berakhir saat memiliki instans yang sedang berjalan, instans dihentikan.

Untuk menambahkan Blok Kapasitas instans ke Grup Sumber Daya Reservasi Kapasitas, lihatGrup Sumber Daya Reservasi Kapasitas.

Untuk membuat Grup Sumber Daya Reservasi UltraServer Kapasitas untuk Blok Kapasitas, gunakan salah satu metode berikut.

AWS CLI

Cara membuat Grup Sumber Daya Reservasi Kapasitas untuk Blo UltraServer k Kapasitas

Gunakan AWS CLI perintah create-group, dan for`--configuration`, tentukan yang berikut ini:

```
{
  "Configuration": [
    {
      "Type": "AWS::EC2::CapacityReservationPool",
      "Parameters": [
        {
          "Name": "instance-type",
          "Values": [
            "instance_type"
          ]
        },
        {
          "Name": "reservation-type",
          "Values": [
            "capacity-block"
          ]
        }
      ]
    },
    {
      "Type": "AWS::ResourceGroups::Generic",
      "Parameters": [
        {
          "Name": "allowed-resource-types",
          "Values": [
            "AWS::EC2::CapacityReservation"
          ]
        }
      ]
    }
  ]
}
```

PowerShell

Cara membuat Grup Sumber Daya Reservasi Kapasitas untuk Blo UltraServer k Kapasitas

Gunakan New-RGGroup cmdlet. Untuk`-Configuration`, tentukan hal berikut:

```
{
  "Configuration": [
    {
```

```
      "Type": "AWS::EC2::CapacityReservationPool",
      "Parameters": [
        {
          "Name": "instance-type",
          "Values": [
            "instance_type"
          ]
        },
        {
          "Name": "reservation-type",
          "Values": [
            "capacity-block"
          ]
        }
      ]
    },
    {
      "Type": "AWS::ResourceGroups::Generic",
      "Parameters": [
        {
          "Name": "allowed-resource-types",
          "Values": [
            "AWS::EC2::CapacityReservation"
          ]
        }
      ]
    }
  ]
}
```

> **Note**
>
> Untuk menambahkan berbagai jenis Reservasi Kapasitas ke UltraServer-only grup yang ada, Anda harus terlebih dahulu memodifikasi konfigurasi grup menggunakan API put-group-configuration dan menghapus bagian dan konfigurasi. `instance-type` `reservation-type` Setelah selesai, lihat Grup Sumber Daya Reservasi Kapasitas operasi pengelompokan lainnya.

Setelah Anda membuat Grup Sumber Daya Reservasi Kapas UltraServer itas untuk Blok Kapasitas, gunakan salah satu metode berikut untuk menambahkan Blok UltraServer Kapasitas yang ada ke dalamnya.

AWS CLI

Untuk menambahkan Blok UltraServer Kapasitas ke Grup Sumber Daya Reservasi Kapasitas

Gunakan perintah group-resources. Untuk `--group` menentukan nama Grup Sumber Daya Reservasi Kapasitas yang Anda buat. Untuk`--resource-arns`, tentukan ARN Blok UltraServer Kapasitas yang akan ditambahkan.

```
aws resource-groups group-resources \
--group MyCRGroup \
--resource-arns CapacityReservationArn
```

PowerShell

Untuk menambahkan Blok UltraServer Kapasitas ke Grup Sumber Daya Reservasi Kapasitas

Gunakan Add-RGResource cmdlet. Untuk `-Group` menentukan nama Grup Sumber Daya Reservasi Kapasitas yang Anda buat. Untuk`-ResourceArn` , tentukan ARN Blok UltraServer Kapasitas yang akan ditambahkan.

Contoh berikut menambahkan dua Reservasi Kapasitas ke grup yang ditentukan.

```
Add-RGResource `
-Group MyCRGroup `
-ResourceArn CapacityReservationArn
```

Memantau Blok Kapasitas menggunakan EventBridge

Saat reservasi Blok Kapasitas Anda dimulai, Amazon EC2 akan mengeluarkan acara EventBridge yang menunjukkan kapasitas Anda siap digunakan. Empat puluh menit sebelum reservasi Blok Kapasitas berakhir, Anda menerima EventBridge acara lain yang memberi tahu Anda bahwa setiap instans yang berjalan dalam reservasi akan mulai berakhir dalam 10 menit. Untuk informasi selengkapnya tentang EventBridge acara, lihat Acara Amazon EventBridge .

Peristiwa berikut menyusun peristiwa yang dipancarkan untuk Blok Kapasitas:

### Blok Kapasitas Dikirimkan

Contoh berikut menunjukkan peristiwa untuk Blok Kapasitas Terkirim.

```
{
  "customer_event_id": "[Capacity Reservation Id]-delivered",
  "detail_type": "Capacity Block Delivered",
  "source": "aws.ec2",
  "account": "[Customer Account ID]",
  "time": "[Current time]",
  "resources": [
    "[ODCR ARN]"
  ],
  "detail": {
    "capacity-reservation-id": "[ODCR ID]",
    "end-date": "[ODCR End Date]"
  }
}
```

### Peringatan Kedaluwarsa Blok Kapasitas

Contoh berikut menunjukkan peristiwa untuk Peringatan Kedaluwars Blok Kapasitas.

```
{
  "customer_event_id": "[Capacity Reservation Id]-approaching-expiry",
  "detail_type": "Capacity Block Expiration Warning",
  "source": "aws.ec2",
  "account": "[Customer Account ID]",
  "time": "[Current time]",
  "resources": [
    "[ODCR ARN]"
  ],
  "detail": {
    "capacity-reservation-id": "[ODCR ID]",
    "end-date": "[ODCR End Date]"
  }
}
```

### Peringatan Gangguan Instans Reservasi Kapasitas

Contoh berikut menunjukkan peristiwa untuk Peringatan Gangguan Instans Reservasi Kapasitas EC2.

```
{
    "version": "0",
    "id": "12345678-1234-1234-1234-123456789012",
    "detail_type": "EC2 Capacity Reservation Instance Interruption Warning",
    "source": "aws.ec2",
    "account": "[Customer Account ID]",
    "time": "[Current time]",
    "region": "[Region]",
    "resources": [
        "[Instance ARN]"
    ],
    "detail": {
        "instance-id": "[Instance ID]",
        "instance-action": "terminate",
        "instance-termination-time": "[Current time]",
        "availability-zone-id": "[Availability Zone ID]",
        "instance-lifecycle": "capacity-block"
    }
}
```

Pencatatan Kapasitas Memblokir panggilan API dengan AWS CloudTrail

Blok Kapasitas terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di Blok Kapasitas. CloudTrail menangkap panggilan API untuk Blok Kapasitas sebagai peristiwa. Panggilan yang ditangkap tersebut mencakup panggilan dari konsol Blok Kapasitas dan panggilan kode ke operasi Blok Kapasitas. Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara terus menerus ke bucket Amazon S3, termasuk peristiwa untuk Blok Kapasitas. Jika Anda tidak membuat konfigurasi jejak, Anda masih dapat melihat kejadian terbaru dalam konsol CloudTrail di Riwayat peristiwa. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke Blok Kapasitas, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan itu dibuat, dan detail tambahan.

Untuk mempelajari selengkap CloudTrail nya, lihat Panduan AWS CloudTrail Pengguna.

Informasi Blok Kapasitas di CloudTrail

CloudTrail diaktifkan pada Anda Akun AWS saat Anda membuat akun. Ketika aktivitas terjadi di Blok Kapasitas, aktivitas tersebut dicatat dalam suatu CloudTrail peristiwa bersama dengan peristiwa AWS layanan lainnya di ri wayat acara. Anda dapat melihat, mencari, dan mengunduh acara terbaru di Akun AWS. Untuk informasi selengkapnya, lihat Melihat acara dengan CloudTrail riwayat acara.

Untuk catatan peristiwa yang berkelanjutan di Anda Akun AWS, termasuk acara untuk Blok Kapasitas, buat jejak. Jej ak CloudTrail memungkinkan pengiriman file log ke bucket Amazon S3. Secara default, saat Anda membuat jejak di konsol, jejak tersebut berlaku untuk semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah di partisi AWS dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan bertindak berdasarkan data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- Gambaran umum untuk membuat jejak
- CloudTrail layanan dan integrasi yang didukung
- Mengkonfigurasi notifikasi Amazon SNS untuk CloudTrail
- M CloudTrail enerima file log dari beberapa wilayah dan Mener CloudTrail ima file log dari beberapa akun

Semua tindakan Blok Kapasitas dicatat oleh CloudTrail dan didokumentasikan dalam Referensi API Amazon EC2. Misalnya, panggilan ke`CapacityBlockScheduled`, dan `CapacityBlockActive` tindakan menghasilkan entri dalam file CloudTrail log.

Setiap entri peristiwa atau log berisi informasi tentang entitas yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut ini:

- Apakah permintaan dibuat dengan kredenSIAL pengguna root atau AWS Identity and Access Management (IAM).
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.
- Apakah permintaan itu dibuat oleh AWS layanan lain.

Untuk informasi selengkapnya, lihat Elemen userIdentity CloudTrail .

Memahami entri file log Blok Kapasitas

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Sebuah kejadian mewakili permintaan tunggal dari sumber apa pun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. Berkas log CloudTrail bukan merupakan jejak tumpukan terurut dari panggilan API publik, sehingga tidak muncul dalam urutan tertentu.

Contoh berikut menunjukkan entri CloudTrail log untuk:

- TerminateCapacityBlocksInstances
- CapacityBlockPaymentFailed
- CapacityBlockScheduled
- CapacityBlockActive
- CapacityBlockFailed
- CapacityBlockExpired

> **Note**
>
> Beberapa bidang telah disensor dari contoh untuk privasi data.

TerminateCapacityBlocksInstances


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "TerminateCapacityBlockInstances",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "accountId": "123456789012",
      "type": "AWS::EC2::Instance",
      "ARN": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0"
    }
    {
      "accountId": "123456789012",
```

```
      "type": "AWS::EC2::Instance",
      "ARN": "arn:aws:ec2:us-east-1:123456789012:instance/i-0598c7d356eba48d7"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      }
}
```

### CapacityBlockPaymentFailed

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockPaymentFailed",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "payment-failed"
      }
}
```

### CapacityBlockScheduled


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockScheduled",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "scheduled"
      }
}
```


### CapacityBlockActive


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
```

```
  "eventName": "CapacityBlockActive",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "active"
      }
 }
```

### CapacityBlockFailed

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockFailed",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
```

```
      "ARN": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "failed"
      }
 }
```

### CapacityBlockExpired

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "accountId": "123456789012",
    "invokedBy": "AWS Internal;"
  },
  "eventTime": "2023-10-02T00:06:08Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CapacityBlockExpired",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "203.0.113.25",
  "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
  "requestParameters": null,
  "responseElements": null,
  "eventID": "a1b2c3d4-EXAMPLE",
  "readOnly": false,
  "resources": [
    {
      "ARN": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-12345678",
      "accountId": "123456789012",
      "type": "AWS::EC2::CapacityReservation"
    }
  ],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "123456789012",
  "serviceEventDetails": {
      "capacityReservationId": "cr-12345678",
      "capacityReservationState": "expired"
```

```
        }
    }
```

## Grup Sumber Daya Reservasi Kapasitas

Anda dapat menggunakan AWS Resource Groups untuk membuat koleksi logis Reservasi Kapasitas, yang disebut Grup Sumber Daya Reservasi Kapasitas (grup). Grup Sumber Daya Reservasi Kapasitas adalah grup sumber daya yang berisi Reservasi Kapasitas Anda dan memungkinkan Anda meluncurkan instans ke beberapa Reservasi Kapasitas dengan menentukan ARN Grup Sumber Daya Reservasi Kapasitas tunggal. Untuk informasi selengkapnya tentang Grup Sumber Daya Reservasi Kapasitas, lihat Apa itu grup sumber daya? dalam Panduan AWS Resource Groups Pengguna.

Saat Anda menentukan grup untuk meluncurkan instans, Amazon EC2 mencocokkan instans dengan Reservasi Kapasitas apa pun dalam grup yang memiliki atribut yang cocok dan kapasitas yang tersedia.

Anda dapat menyertakan Reservasi Kapasitas yang Anda miliki di akun Anda, dan Reservasi Kapasitas yang dibagikan dengan Anda oleh AWS akun lain dalam satu grup. Anda juga dapat menyertakan Reservasi Kapasitas yang memiliki atribut berbeda (jenis instans, platform, Zona Ketersediaan, sewa, dan grup penempatan) dalam satu grup.

Grup mendukung semua jenis reservasi, termasuk Reservasi Kapasitas, Reserv asi On-Demand Kapasitas yang dapat diinterupsi, dan Blo k Kapasitas untuk ML. Anda dapat menambahkan kombinasi jenis reservasi ke satu grup.

Tugas

- Membuat grup
- Menambahkan Reservasi Kapasitas ke grup
- Meluncurkan instance ke Reservasi Kapasitas dalam grup
- Menghapus Reservasi Kapasitas dari grup
- Siklus hidup instans dalam grup
- Menghapus grup

### Membuat grup

Anda dapat membuat grup untuk mengatur Reservasi Kapasitas Anda. Setelah membuat grup, Anda dapat menambahkan Reservasi Kapasitas ke grup.

Console

Untuk membuat grup

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Grup  Sumber Daya Reservasi Kapasitas.
4. Pilih Buat group.
5. Masukkan nama grup, dan secara opsional tambahkan deskripsi, Reservasi Kapasitas, dan tag.
6. Pilih Buat.

AWS CLI

Untuk membuat grup untuk Reservasi Kapasitas

Gunakan AWS CLI perintah create-group dengan parameter permintaan berikut.

- `AWS::EC2::CapacityReservationPool`- Memastikan bahwa Grup Sumber Daya Reservasi Kapasitas dapat ditargetkan misalnya peluncuran.
- `AWS::ResourceGroups::Generic`dengan dis `allowed-resource-types` etel ke `AWS::EC2::CapacityReservation` — Memastikan bahwa Grup Sumber Daya Reservasi Kapasitas hanya menerima Reservasi Kapasitas.

```
aws resource-groups create-group \
    --name MyCRGroup \
    --configuration \
        '{"Type": "AWS::EC2::CapacityReservationPool"}' \
        '{"Type": "AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-
resource-types", "Values": ["AWS::EC2::CapacityReservation"]}]}'
```

PowerShell

Untuk membuat grup untuk Reservasi Kapasitas

Gunakan New-RGGroup cmdlet.

```
New-RGGroup `
```

```
    -Name MyCRGroup `
    -Configuration `
        @{"Type"="AWS::EC2::CapacityReservationPool"} `
        @{"Type"="AWS::ResourceGroups::Generic"; "Parameters"=@{"allowed-resource-
types"=@{"Values"="AWS::EC2::CapacityReservation"}}}
```

Untuk membuat grup yang hanya menerima Blo UltraServer k Kapasitas dari jenis instans tertentu, lihatMembuat Grup Sumber Daya Reservasi Kapas UltraServer itas untuk Blok Kapasitas.

Menambahkan Reservasi Kapasitas ke grup

Anda dapat menambahkan Reservasi Kapasitas yang Anda miliki di akun Anda, atau Reservasi Kapasitas yang dibagikan dengan Anda oleh AWS akun lain ke grup.

Pertimbangan-pertimbangan

- Anda hanya dapat menambahkan Reservasi Kapasitas di `active` negara bagian ke grup.
- Jika Reservasi Kapasitas bersama kemudian tidak dibagikan oleh pemiliknya, Amazon EC2 akan menghapusnya secara otomatis dari grup Anda.

Console

Anda dapat menambahkan Reservasi Kapasitas ke grup dari halaman detail Grup Sumber Daya Reservasi Kapasitas, atau dari konsol Reservasi Kapasitas.

Untuk menambahkan Reservasi Kapasitas dari halaman Grup Sumber Daya Reservasi Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Grup Sumber Daya Reservasi Kapasitas.
4. Pilih nama grup untuk membuka halaman detailnya.
5. Di bagian Reserv asi Kapasitas, pilih Tam bah.
6. Pilih Reservasi Kapasitas yang ingin Anda tambahkan, lalu pilih Tam bah.

Untuk menambahkan Reservasi Kapasitas dari konsol Reservasi Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Reservasi Kapasitas yang ingin Anda tambahkan ke grup.
4. Pilih Tind akan, lalu pilih Tambahkan ke grup.
5. Pilih grup, lalu pilih Tam bah.

AWS CLI

Untuk menambahkan Reservasi Kapasitas ke grup

Gunakan perintah group-resources. Berikan ARN dari setiap Reservasi Kapasitas yang ingin Anda tambahkan.

Contoh berikut menambahkan dua Reservasi Kapasitas ke grup yang ditentukan.

```
aws resource-groups group-resources \
    --group MyCRGroup \
    --resource-arns \
        arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1
 \
        arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

PowerShell

Untuk menambahkan Reservasi Kapasitas ke grup

Gunakan Add-RGResource cmdlet.

Contoh berikut menambahkan dua Reservasi Kapasitas ke grup yang ditentukan.

```
Add-RGResource `
    -Group MyCRGroup `
    -ResourceArn `
        "arn:aws:ec2:sa-east-1:123456789012:capacity-
reservation/cr-1234567890abcdef1", `
        "arn:aws:ec2:sa-east-1:123456789012:capacity-
reservation/cr-54321abcdef567890"
```

Meluncurkan instance ke Reservasi Kapasitas dalam grup

Anda dapat meluncurkan instans ke Reservasi Kapasitas yang ada dengan menentukan Grup Sumber Daya Reservasi Kapasitas ARN. Instans yang menargetkan grup cocok dengan Reservasi Kapasitas apa pun dalam grup yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa) dan kapasitas yang tersedia.

Saat meluncurkan instans menggunakan wizard instance peluncuran atau RunInstances API, Anda hanya dapat menentukan satu opsi pembelian (satu jenis reservasi) per peluncuran. Untuk meluncurkan ke beberapa jenis reservasi dalam grup sekaligus, gunakan grup Amazon EC2 Fleet atau Auto Scaling.

Untuk panduan langkah demi langkah tentang cara meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas, lihat tutorial berikut: melalui Armada EC2 dan melalui Amazon EC2 Auto Scaling.

On-Demand Reservasi Kapasitas

Untuk meluncurkan instans ke Reservasi On-Demand Kapasitas dalam grup, tentukan grup ARN menggunakan `CapacityReservationTarget` parameter dengan `CapacityReservationResourceGroupArn` bidang.

Amazon EC2 meluncurkan instans ke Reservasi On-Demand Kapasitas yang cocok di grup. Jika tidak ada kapasitas pencocokan yang tersedia, instans diluncurkan menggunakan On-Demand kapasitas.

Jika Anda tidak ingin instance diluncurkan seperti On-Demand ketika tidak ada kapasitas yang cocok tersedia, setel `CapacityReservationPreference` ke`capacity-reservations-only`. Dengan pengaturan ini, instans gagal diluncurkan jika tidak ada Reservasi On-Demand Kapasitas yang cocok dalam grup yang memiliki kapasitas yang tersedia.

Console

Untuk meluncurkan instans ke On-Demand Reservasi Kapasitas dalam grup

1. Ikuti prosedur untuk meluncurkan instance, tetapi jangan luncurkan instans sampai Anda menyelesaikan langkah-langkah berikut.
2. Perluas detail lanjutan.
3. Untuk opsi Pembel ian, biarkan sebagai Tidak Ada (default).

4. Untuk Reservasi Kapasitas, pilih Tentukan Grup Sumber Daya Reservasi Kapasitas (default) atau Hanya luncurkan di Grup Sumber Daya Reservasi Kapasitas (untuk mencegah On-Demand fallback), lalu pilih grup.
5. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan instans ke On-Demand Reservasi Kapasitas dalam grup

Gunakan perintah run-instance dengan parameter. `--capacity-reservation-specification`

Contoh berikut meluncurkan instance ke dalam grup. Jika grup tidak memiliki ODCR dengan atribut yang cocok dan kapasitas yang tersedia, instans diluncurkan sebagai On-Demand instance.

```
aws ec2 run-instances \
    --instance-type c5.xlarge \
    --image-id ami-0abcdef1234567890 \
    --count 3 \
    --capacity-reservation-specification \

 CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup}
```

Contoh berikut meluncurkan instance ke dalam grup. Jika grup tidak memiliki ODCR dengan atribut yang cocok dan kapasitas yang tersedia, instans gagal diluncurkan.

```
aws ec2 run-instances \
    --instance-type c5.xlarge \
    --image-id ami-0abcdef1234567890 \
    --count 3 \
    --capacity-reservation-specification \
        CapacityReservationPreference=capacity-reservations-only,CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup}
```

### Menargetkan grup penempatan tertentu

Secara opsional, jika grup Anda berisi On-Demand Reservasi Kapasitas dalam grup penempatan, Anda juga dapat menentukan ID grup penempatan bersama dengan ARN Grup Sumber Daya Reservasi Kapasitas untuk menempatkan semua instans yang diluncurkan ke dalam grup penempatan tersebut.

> **Note**
>
> Blok Kapasitas tidak mendukung grup penempatan. Anda tidak dapat menentukan grup penempatan saat permintaan menargetkan Blok Kapasitas dalam Grup Sumber Daya Reservasi Kapasitas.

## Reservasi Kapasitas yang Dapat Diinterupsi

Untuk meluncurkan instans ke Reservasi Kapasitas yang dapat diinterupsi dalam grup, tentukan grup ARN dan pilih opsi pembelian yang dapat diinterupsi. Amazon EC2 menempatkan instans ke dalam Reservasi Kapasitas aktif yang dapat diinterupsi dalam grup dengan atribut yang cocok dan kapasitas yang tersedia. Jika tidak ada kapasitas pencocokan yang tersedia, instans gagal diluncurkan.

### Console

Untuk meluncurkan instans ke Reservasi Kapasitas yang dapat diinterupsi dalam grup

1. Ikuti prosedur untuk meluncurkan instance, tetapi jangan luncurkan instans sampai Anda menyelesaikan langkah-langkah berikut.
2. Perluas detail  lanjutan.
3. Untuk opsi  Pembelian, pilih Reserv  asi Kapasitas Terputus.
4. Untuk  Reservasi Kapasitas, pilih  Tentukan Grup Sumber Daya Reservasi Kapasitas  dan pilih grup.
5. Pilih Luncurkan instans.

### AWS CLI

Untuk meluncurkan instans ke Reservasi Kapasitas yang dapat diinterupsi dalam grup

Gunakan perintah run-instance dengan parameter `--instance-market-options` dan `--capacity-reservation-specification`.

```
aws ec2 run-instances \
    --instance-type c5.xlarge \
    --image-id ami-0abcdef1234567890 \
    --count 3 \
    --instance-market-options MarketType=interruptible-capacity-reservation \
    --capacity-reservation-specification \

 CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup}
```

Blok Kapasitas

Untuk meluncurkan instance ke Blok Kapasitas dalam grup, tentukan grup ARN dan pilih opsi pembelian Blok Kapasitas. Amazon EC2 menempatkan instance ke Blok Kapasitas aktif apa pun di grup dengan atribut yang cocok dan kapasitas yang tersedia. Jika tidak ada kapasitas pencocokan yang tersedia, instans gagal diluncurkan.

> **Note**
>
> Anda tidak dapat menentukan grup penempatan saat meluncurkan ke Blok Kapasitas di Grup Sumber Daya Reservasi Kapasitas.

Console

Untuk meluncurkan instance ke Blok Kapasitas dalam grup

1. Ikuti prosedur untuk meluncurkan instance, tetapi jangan luncurkan instans sampai Anda menyelesaikan langkah-langkah berikut.
2. Perluas detail lanjutan.
3. Untuk opsi Pembelian, pilih Blo k Kapasitas.
4. Untuk Reservasi Kapasitas, pilih Tentukan Grup Sumber Daya Reservasi Kapasitas dan pilih grup.
5. Pilih Luncurkan instans.

AWS CLI

Untuk meluncurkan instance ke Blok Kapasitas dalam grup

Gunakan perintah run-instance dengan parameter `--instance-market-options` dan `--capacity-reservation-specification`.

```
aws ec2 run-instances \
    --instance-type p5.48xlarge \
    --image-id ami-0abcdef1234567890 \
    --count 1 \
    --instance-market-options MarketType=capacity-block \
    --capacity-reservation-specification \

 CapacityReservationTarget={CapacityReservationResourceGroupArn=arn:aws:resource-groups:sa-east-1:123456789012:group/MyCRGroup}
```

Menghapus Reservasi Kapasitas dari grup

Anda dapat menghapus Reservasi Kapasitas dari grup kapan saja. Instans yang menargetkan grup terus berjalan di Reservasi Kapasitas saat ini dan tidak terpengaruh oleh penghapusan. Reservasi Kapasitas mengembalikan kapasitas hanya saat Anda menghentikan instans.

> **Note**
>
> Jika Reservasi Kapasitas yang dibagikan dengan Anda nantinya tidak dibagikan, Amazon EC2 akan menghapusnya secara otomatis dari grup.

Console

Untuk menghapus Reservasi Kapasitas dari grup

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Grup  Sumber Daya Reservasi Kapasitas.
4. Pilih nama grup untuk membuka halaman detailnya.
5. Di  bagian Reserv  asi Kapasitas, pilih Reservasi Kapasitas yang akan dihapus, lalu pilih H apus.

AWS CLI

Untuk menghapus Reservasi Kapasitas dari grup

Gunakan perintah ungroup-resources.

Contoh berikut menghapus dua Reservasi Kapasitas dari grup yang ditentukan.

```
aws resource-groups ungroup-resources \
    --group MyCRGroup \
    --resource-arns \
        arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-0e154d26a16094dd
 \
        arn:aws:ec2:sa-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

PowerShell

Untuk menghapus Reservasi Kapasitas dari grup

Gunakan Remove-RGResource cmdlet.

Contoh berikut menghapus dua Reservasi Kapasitas dari grup yang ditentukan.

```
Remove-RGResource `
    -Group MyCRGroup `
    -ResourceArn `
        "arn:aws:ec2:sa-east-1:123456789012:capacity-
reservation/cr-0e154d26a16094dd", `
        "arn:aws:ec2:sa-east-1:123456789012:capacity-
reservation/cr-54321abcdef567890"
```

Siklus hidup instans dalam grup

Ketika Reservasi Kapasitas dalam grup berubah status, dampaknya terhadap instans yang sedang berjalan tergantung pada jenis reservasi.

On-Demand Reservasi Kapasitas

Ketika Reservasi On-Demand Kapasitas dalam grup dibatalkan, kedaluwarsa, atau kapasitasnya berkurang, instans yang menargetkan grup cocok dengan Reservasi On-Demand Kapasitas lainnya

dalam grup yang memiliki atribut yang cocok (jenis instans, platform, Zona Ketersediaan, dan sewa) dan kapasitas yang tersedia. Jika grup tidak memiliki Reservasi On-Demand Kapasitas dengan atribut yang cocok dan kapasitas yang tersedia, instans dijalankan menggunakan On-Demand kapasitas. Jika Anda menambahkan Reservasi On-Demand Kapasitas yang cocok ke grup nanti, Amazon EC2 secara otomatis memindahkan instans ke kapasitas yang dicadangkan.

Reservasi Kapasitas yang Dapat Diinterupsi

Ketika kapasitas Reservasi Kapasitas yang dapat diinterupsi direklamasi kembali, instans menerima pemberitahuan gangguan 2 menit sebelum penghentian. Untuk informasi selengkapnya, lihat Pengalaman gangguan.

Blok Kapasitas

Instans yang berjalan di Blok Kapasitas di Grup Sumber Daya Reservasi Kapasitas dihentikan sebelum Blok Kapasitas berakhir, sama seperti untuk Blok Kapasitas yang Anda targetkan secara langsung. Untuk informasi selengkapnya, lihat Cara kerja Blok Kapasitas Amazon EC2.

Menghapus grup

Jika Anda tidak lagi membutuhkan grup Anda, Anda dapat menghapusnya kapan saja. Saat Anda menghapus grup dengan instans yang sedang berjalan dan Reservasi Kapasitas aktif, instans yang menargetkan grup akan terus berjalan di Reservasi Kapasitas saat ini setelah penghapusan grup. Amazon EC2 mengembalikan kapasitas ke Reservasi Kapasitas saat Anda menghentikan instans.

Console

Untuk menghapus grup

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Reservasi Kapasitas.
3. Pilih Grup Sumber Daya Reservasi Kapasitas.
4. Pilih grup yang ingin Anda hapus, pilih T indakan, lalu pilih H apus.
5. Saat diminta untuk mengonfirmasi, pilih H apus.

AWS CLI

Untuk menghapus grup

Gunakan perintah delete-group.

```
aws resource-groups delete-group --group MyCRGroup
```

PowerShell

Untuk menghapus grup

Gunakan Remove-RGGroup cmdlet.

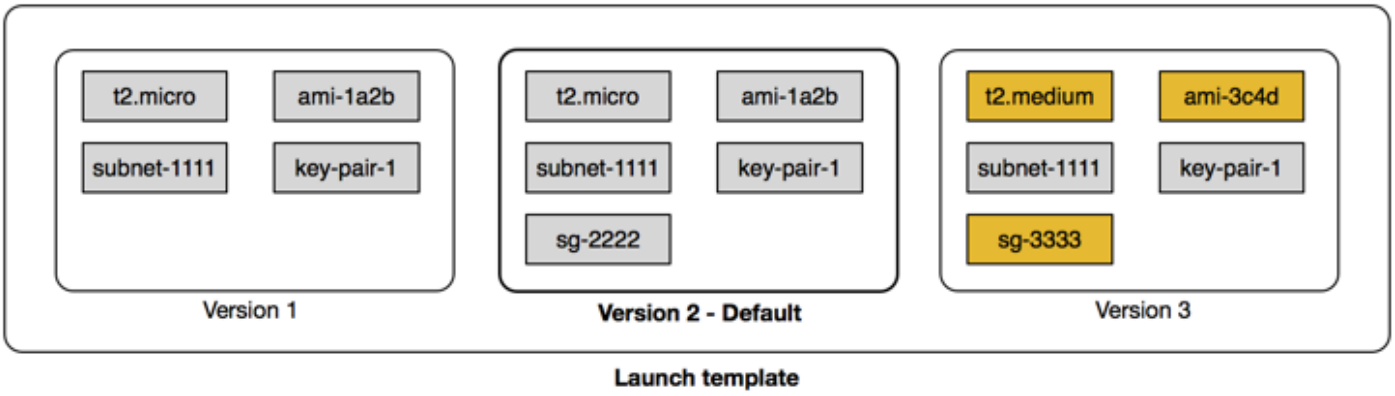
```
Remove-RGGroup -GroupName MyCRGroup
```

# Simpan parameter peluncuran instans di templat peluncuran Amazon EC2

Anda dapat menggunakan template peluncuran Amazon EC2 untuk menyimpan parameter peluncuran instans sehingga Anda tidak perlu menentukannya setiap kali meluncurkan instans Amazon EC2. Misalnya, Anda dapat membuat template peluncuran yang menyimpan ID AMI, jenis instans, dan pengaturan jaringan yang biasanya Anda gunakan untuk meluncurkan instans. Saat meluncurkan instance menggunakan konsol Amazon EC2, AWS SDK, atau alat baris perintah, Anda dapat menentukan template peluncuran alih-alih memasukkan parameter lagi.

Untuk setiap templat peluncuran, Anda dapat membuat satu atau beberapa versi templat peluncuran bernomor. Setiap versi dapat memiliki parameter peluncuran yang berbeda. Saat Anda meluncurkan sebuah instans dari templat peluncuran, Anda dapat menggunakan templat peluncuran versi apa pun. Jika Anda tidak menentukan versi, versi default digunakan. Anda dapat mengatur templat peluncuran versi apa pun sebagai versi default—secara default, ini adalah templat peluncuran versi pertama.

Diagram berikut memperlihatkan templat peluncuran dengan tiga versi. Versi pertama menentukan tipe instans, ID AMI, subnet, dan pasangan kunci yang akan digunakan untuk meluncurkan instans. Versi kedua didasarkan pada versi pertama dan juga menentukan grup keamanan untuk instans tersebut. Versi ketiga menggunakan nilai yang berbeda untuk beberapa parameter. Versi 2 ditetapkan sebagai versi default. Jika Anda meluncurkan sebuah instans dari templat peluncuran ini, parameter peluncuran dari versi 2 akan digunakan jika tidak ada versi lain yang ditentukan.




Daftar Isi


- Pembatasan untuk templat peluncuran Amazon EC2
- Izin IAM diperlukan untuk templat peluncuran Amazon EC2
- Gunakan templat peluncuran Amazon EC2 untuk mengontrol peluncuran instans Amazon EC2
- Buat template peluncuran Amazon EC2
- Modifikasi templat peluncuran (mengelola versi templat peluncuran)
- Menghapus template peluncuran atau versi template peluncuran


## Pembatasan untuk templat peluncuran Amazon EC2

Batasan berikut berlaku untuk peluncuran template dan peluncuran versi template:

- Kuota — Untuk melihat kuota templat peluncuran dan versi template peluncuran, buka konsol Kuota Layanan atau gunakan perintah list-service- AWS CLI . Setiap AWS akun dapat memiliki maksimal 5.000 template peluncuran per Wilayah dan hingga 10.000 versi per template peluncuran. Akun Anda mungkin memiliki kuota yang berbeda berdasarkan usia dan riwayat penggunaannya.
- Parameter bersifat opsional - Parameter templat peluncuran bersifat opsional. Namun, Anda harus memastikan bahwa permintaan peluncuran instans menyertakan semua parameter yang diperlukan. Misalnya, jika template peluncuran Anda tidak menyertakan ID AMI, Anda harus menentukan ID AMI saat meluncurkan instance dengan template peluncuran ini.
- Parameter tidak divalidasi — Parameter templat peluncuran tidak sepenuhnya divalidasi saat Anda membuat templat peluncuran. Jika Anda menentukan nilai yang salah atau menggunakan kombinasi parameter yang tidak didukung, instance akan gagal diluncurkan menggunakan template peluncuran ini. Untuk menghindari masalah, pastikan untuk menentukan nilai yang benar

dan gunakan kombinasi parameter yang didukung. Misalnya, untuk meluncurkan sebuah instans di grup penempatan, Anda harus menentukan tipe instans yang didukung.

- Tag — Anda dapat menandai template peluncuran, tetapi Anda tidak dapat menandai versi template peluncuran.
- Tidak dapat diubah - Templat peluncuran tidak dapat diubah. Untuk memodifikasi templat peluncuran, Anda harus membuat templat peluncuran versi baru.
- Nomor versi – Versi templat peluncuran diberi nomor sesuai urutan pembuatannya. Saat membuat versi template peluncuran, Anda tidak dapat menentukan sendiri nomor versi.

## Izin IAM diperlukan untuk templat peluncuran Amazon EC2

Anda dapat menggunakan izin IAM untuk mengontrol apakah pengguna dapat membuat daftar, melihat, membuat, atau menghapus templat peluncuran atau meluncurkan versi template.

> **Important**
>
> Anda tidak dapat menggunakan izin tingkat sumber daya untuk membatasi sumber daya yang dapat ditentukan pengguna dalam template peluncuran saat mereka membuat template peluncuran atau meluncurkan versi template. Oleh karena itu, pastikan hanya administrator tepercaya yang diberikan izin untuk membuat template peluncuran dan meluncurkan versi template.

Anda harus memberi siapa pun yang akan menggunakan template peluncuran izin yang diperlukan untuk membuat dan mengakses sumber daya yang ditentukan dalam template peluncuran. Contoh:

- Untuk meluncurkan instance dari Amazon Machine Image (AMI) pribadi bersama, pengguna template peluncuran harus memiliki izin peluncuran untuk AMI.
- Untuk membuat volume EBS dengan tag dari snapshot yang ada, pengguna template peluncuran harus memiliki akses baca ke snapshot, dan izin untuk membuat dan menandai volume.

Daftar Isi


- ec2: CreateLaunchTemplate
- ec2: DescribeLaunchTemplates
- ec2: DescribeLaunchTemplateVersions

- ec2: DeleteLaunchTemplate
- Mengontrol izin versioning
- Kontrol akses ke tanda pada templat peluncuran

## ec2: CreateLaunchTemplate

Untuk membuat templat peluncuran di konsol atau dengan menggunakan API, pengguna utama harus memiliki izin `ec2:CreateLaunchTemplate` dalam kebijakan IAM. Kapan pun memungkinkan, gunakan tanda untuk membantu Anda mengontrol akses ke templat peluncuran di akun Anda.

Misalnya, pernyataan kebijakan IAM berikut memberikan izin kepada pengguna utama untuk membuat templat peluncuran hanya jika templat menggunakan tanda yang ditentukan (*purpose*=*testing*).

```
{
    "Sid": "IAMPolicyForCreatingTaggedLaunchTemplates",
    "Action": "ec2:CreateLaunchTemplate",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/purpose": "testing"
        }
    }
}
```

Pengguna utama yang membuat templat peluncuran mungkin memerlukan beberapa izin terkait, seperti:

- ec2: CreateTags — Untuk menambahkan tag ke template peluncuran selama `CreateLaunchTemplate` operasi, pemang `CreateLaunchTemplate` gil harus memiliki `ec2:CreateTags` izin dalam kebijakan IAM.
- ec2: RunInstances — Untuk meluncurkan instans EC2 dari template peluncuran yang mereka buat, prinsipal juga harus memiliki `ec2:RunInstances` izin dalam kebijakan IAM.

Untuk tindakan-tindakan yang digunakan untuk membuat sumber daya yang menerapkan tanda, para pengguna harus memiliki izin `ec2:CreateTags`. Pernyataan kebijakan IAM berikut menggunakan

kunci syarat `ec2:CreateAction` agar para pengguna dapat membuat tanda hanya dalam konteks `CreateLaunchTemplate`. Pengguna tidak dapat menandai templat peluncuran yang ada atau sumber daya lainnya. Untuk informasi selengkapnya, lihat Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan.

```
{
    "Sid": "IAMPolicyForTaggingLaunchTemplatesOnCreation",
    "Action": "ec2:CreateTags",
    "Effect": "Allow",
    "Resource": "arn:aws:ec2:us-east-1:111122223333:launch-template/*",
    "Condition": {
        "StringEquals": {
            "ec2:CreateAction": "CreateLaunchTemplate"
        }
    }
}
```

Pengguna IAM yang membuat templat peluncuran tidak secara otomatis memiliki izin untuk menggunakan templat peluncuran yang mereka buat. Seperti pengguna utama lainnya, pembuat templat peluncuran perlu mendapatkan izin melalui kebijakan IAM. Jika pengguna IAM ingin meluncurkan instans EC2 dari templat peluncuran, mereka harus memiliki izin `ec2:RunInstances`. Saat memberikan izin ini, Anda dapat menentukan bahwa pengguna hanya dapat menggunakan templat peluncuran dengan tanda tertentu atau ID tertentu. Anda juga dapat mengontrol AMI dan sumber daya lain yang dapat dirujuk dan digunakan oleh siapa pun yang menggunakan templat peluncuran saat meluncurkan instans dengan menentukan izin tingkat sumber daya untuk panggilan `RunInstances` tersebut. Untuk kebijakan-kebijakan contoh, lihat Templat peluncuran.

### ec2: DescribeLaunchTemplates

Untuk membuat daftar dan melihat template peluncuran di akun, kepala sekolah harus memiliki `ec2:DescribeLaunchTemplates` izin dalam kebijakan IAM. Karena tindakan `Describe` tidak mendukung izin tingkat sumber daya, Anda harus menentukannya tanpa syarat dan nilai elemen sumber daya dalam kebijakan harus `"*"`.

Misalnya, pernyataan kebijakan IAM berikut memberikan izin utama untuk mencantumkan dan melihat semua templat peluncuran di akun.

```
{
    "Sid": "IAMPolicyForDescribingLaunchTemplates",
    "Action": "ec2:DescribeLaunchTemplates",
```

```
    "Effect": "Allow",
    "Resource": "*"
}
```

### ec2: DescribeLaunchTemplateVersions

Kepala sekolah yang membuat daftar dan melihat template peluncuran juga harus memiliki `ec2:DescribeLaunchTemplateVersions` izin untuk mengambil seluruh rangkaian atribut yang membentuk template peluncuran.

Untuk membuat daftar dan melihat versi template peluncuran di akun, kepala sekolah harus memiliki `ec2:DescribeLaunchTemplateVersions` izin dalam kebijakan IAM. Karena tindakan `Describe` tidak mendukung izin tingkat sumber daya, Anda harus menentukannya tanpa syarat dan nilai elemen sumber daya dalam kebijakan harus `"*"`.

Misalnya, pernyataan kebijakan IAM berikut memberikan izin utama untuk mencantumkan dan melihat semua versi template peluncuran di akun.

```
{
    "Sid": "IAMPolicyForDescribingLaunchTemplateVersions",
    "Effect": "Allow",
    "Action": "ec2:DescribeLaunchTemplateVersions",
    "Resource": "*"
}
```

### ec2: DeleteLaunchTemplate

> **Important**
>
> Berhati-hatilah saat memberikan izin kepada pengguna utama untuk menghapus sumber daya. Menghapus template peluncuran dapat menyebabkan kegagalan dalam AWS sumber daya yang bergantung pada template peluncuran.

Untuk menghapus templat peluncuran, pengguna utama harus memiliki izin `ec2:DeleteLaunchTemplate` dalam kebijakan IAM. Sebisa mungkin, gunakan kunci syarat berbasis tanda untuk membatasi izin.

Misalnya, pernyataan kebijakan IAM berikut memberikan izin utama untuk menghapus template peluncuran hanya jika template memiliki tag (*`purpose`*=*`testing`*) yang ditentukan.

```
{
    "Sid": "IAMPolicyForDeletingLaunchTemplates",
    "Action": "ec2:DeleteLaunchTemplate",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/purpose": "testing"
        }
    }
}
```

Atau, Anda dapat menggunakan ARNs untuk mengidentifikasi templat peluncuran yang menerapkan kebijakan IAM.

Template peluncuran memiliki ARN berikut.

```
"Resource": "arn:aws:ec2:us-east-1:111122223333:launch-template/lt-09477bcd97b0d310e"
```

Anda dapat menentukan banyak ARN dengan memasukkannya dalam suatu daftar, atau Anda dapat menentukan nilai `Resource` dari `"*"` tanpa elemen `Condition` agar pengguna utama dapat menghapus versi templat peluncuran apa pun di akun.

## Mengontrol izin versioning

Untuk administrator tepercaya, Anda dapat memberikan akses untuk membuat dan menghapus versi templat peluncuran, dan untuk mengubah versi default templat peluncuran, dengan menggunakan kebijakan IAM yang mirip dengan contoh berikut.

**Important**

Berhati-hatilah saat memberikan izin kepada kepala sekolah untuk membuat versi template peluncuran atau memodifikasi template peluncuran.

- Saat membuat versi template peluncuran, Anda memengaruhi AWS sumber daya apa pun yang memungkinkan Amazon EC2 meluncurkan instans atas nama Anda dengan `Latest` versi tersebut.
- Saat Anda memodifikasi template peluncuran, Anda dapat mengubah versi mana yang merupakan `Default` dan karenanya memengaruhi AWS sumber daya apa pun yang

memungkinkan Amazon EC2 meluncurkan instans atas nama Anda dengan versi yang dimodifikasi ini.

Anda juga harus berhati-hati dalam menangani AWS sumber daya yang berinteraksi dengan `Latest` atau `Default` meluncurkan versi template, seperti EC2 Fleet dan Spot Fleet. Ketika versi templat peluncuran yang berbeda digunakan `Latest` atau `Default`, Amazon EC2 tidak memeriksa kembali izin untuk tindakan yang harus diselesaikan saat meluncurkan instans baru untuk memenuhi kapasitas target armada karena tidak ada interaksi pengguna dengn sumber daya AWS . Dengan memberikan izin kepada pengguna untuk memanggil API `CreateLaunchTemplateVersion` dan `ModifyLaunchTemplate`, pengguna secara efektif juga diberikan izin `iam:PassRole` jika mereka mengarahkan armada ke versi templat peluncuran berbeda yang berisi profil instans (kontainer untuk peran IAM). Ini berarti bahwa pengguna berpotensi memperbarui template peluncuran untuk meneruskan peran IAM ke instans meskipun mereka tidak memiliki `iam:PassRole` izin. Anda dapat mengelola risiko ini dengan berhati-hati saat memberikan izin kepada siapa yang dapat membuat dan mengelola versi templat peluncuran.

ec2: CreateLaunchTemplateVersion

Untuk menghapus templat peluncuran versi baru, pengguna utama harus memiliki izin `ec2:CreateLaunchTemplateVersion` untuk templat peluncuran dalam kebijakan IAM.

Misalnya, pernyataan kebijakan IAM berikut memberikan izin kepada pengguna utama untuk membuat templat peluncuran hanya jika versi menggunakan tanda yang ditentukan (*`environment`*=*`production`*). Atau, Anda dapat menentukan satu atau banyak ARN templat peluncuran, atau Anda dapat menentukan yang diizinkan `Resource` dari `"*"` tanpa elemen `Condition` agar pengguna utama dapat membuat versi templat peluncuran apa pun di akun.

```
{
    "Sid": "IAMPolicyForCreatingLaunchTemplateVersions",
    "Action": "ec2:CreateLaunchTemplateVersion",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/environment": "production"
        }
    }
```

```
}
```

ec2: DeleteLaunchTemplateVersion

> **Important**
>
> Seperti biasa, Anda harus berhati-hati saat memberikan izin kepada pengguna utama untuk menghapus sumber daya. Menghapus versi template peluncuran dapat menyebabkan kegagalan dalam AWS sumber daya yang bergantung pada versi template peluncuran.

Untuk menghapus versi templat peluncuran, pengguna utama harus memiliki izin `ec2:DeleteLaunchTemplateVersion` untuk templat peluncuran dalam kebijakan IAM.

Misalnya, pernyataan kebijakan IAM berikut memberikan izin kepada pengguna utama untuk menghapus templat peluncuran hanya jika versi menggunakan tanda yang ditentukan (*`environment`*=*`production`*). Atau, Anda dapat menentukan satu atau banyak ARN templat peluncuran, atau Anda dapat menentukan yang diizinkan `Resource` dari `"*"` tanpa elemen `Condition` agar pengguna utama dapat menghapus versi templat peluncuran apa pun di akun.

```
{
    "Sid": "IAMPolicyForDeletingLaunchTemplateVersions",
    "Action": "ec2:DeleteLaunchTemplateVersion",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/environment": "production"
        }
    }
}
```

ec2: ModifyLaunchTemplate

Untuk mengubah versi `Default` yang dikaitkan dengan templat peluncuran, pengguna utama harus memiliki izin `ec2:ModifyLaunchTemplate` untuk templat peluncuran dalam kebijakan IAM.

Misalnya, pernyataan kebijakan IAM berikut memberikan izin kepada pengguna utama untuk memodifikasi templat peluncuran hanya jika templat menggunakan tanda yang ditentukan

(*environment*=*production*). Atau, Anda dapat menentukan satu atau banyak ARN templat peluncuran, atau Anda dapat menentukan yang diizinkan `Resource` dari `"*"` tanpa elemen `Condition` agar pengguna utama dapat mengubah versi templat peluncuran apa pun di akun.

```
{
    "Sid": "IAMPolicyForModifyingLaunchTemplates",
    "Action": "ec2:ModifyLaunchTemplate",
    "Effect": "Allow",
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "aws:ResourceTag/environment": "production"
        }
    }
}
```

## Kontrol akses ke tanda pada templat peluncuran

Anda dapat menggunakan kunci syarat untuk membatasi izin penandaan jika sumber daya adalah templat peluncuran. Misalnya, kebijakan IAM berikut hanya mengizinkan penghapusan tanda dengan kunci *temporary* dari templat peluncuran di akun dan Wilayah yang ditentukan.

```
{
    "Sid": "IAMPolicyForDeletingTagsOnLaunchTemplates",
    "Action": "ec2:DeleteTags",
    "Effect": "Allow",
    "Resource": "arn:aws:ec2:us-east-1:111122223333:launch-template/*",
    "Condition": {
        "ForAllValues:StringEquals": {
            "aws:TagKeys": ["temporary"]
        }
    }
}
```

Untuk informasi selengkapnya tentang kunci syarat yang dapat Anda gunakan untuk mengontrol kunci tanda dan nilai tanda yang dapat diterapkan ke sumber daya Amazon EC2, lihat Mengendalikan akses ke tanda-tanda tertentu.

# Gunakan templat peluncuran Amazon EC2 untuk mengontrol peluncuran instans Amazon EC2

Anda dapat mengontrol konfigurasi instans Amazon EC2 dengan menentukan bahwa pengguna hanya dapat meluncurkan instans jika mereka menggunakan templat peluncuran, dan bahwa mereka hanya dapat menggunakan templat peluncuran tertentu. Anda juga dapat mengontrol siapa yang dapat membuat, memodifikasi, mendeskripsikan, dan menghapus templat peluncuran serta meluncurkan versi templat.

## Gunakan templat peluncuran untuk mengontrol parameter peluncuran

Template peluncuran dapat berisi semua atau beberapa parameter untuk mengkonfigurasi instance saat peluncuran. Namun, saat Anda meluncurkan instance menggunakan template peluncuran, Anda dapat mengganti parameter yang ditentukan dalam template peluncuran. Atau, Anda dapat menentukan parameter tambahan yang tidak ada di templat peluncuran.

> **Note**
>
> Anda tidak dapat menghapus parameter template peluncuran selama peluncuran (misalnya, Anda tidak dapat menentukan nilai null untuk parameter). Untuk menghapus parameter, buat templat peluncuran versi baru tanpa parameter dan gunakan versi tersebut untuk meluncurkan instans.

Untuk meluncurkan instance, pengguna harus memiliki izin untuk menggunakan `ec2:RunInstances` tindakan. Pengguna juga harus memiliki izin untuk membuat atau menggunakan sumber daya yang dibuat atau dikaitkan dengan instans. Anda dapat menggunakan izin tingkat sumber daya untuk tindakan `ec2:RunInstances` untuk mengontrol parameter peluncuran yang dapat ditentukan pengguna. Atau, Anda dapat memberi pengguna izin untuk meluncurkan sebuah instans menggunakan templat peluncuran. Ini memungkinkan Anda untuk mengelola parameter peluncuran di templat peluncuran, bukan di kebijakan IAM, dan untuk menggunakan templat peluncuran sebagai sarana otorisasi untuk meluncurkan instans. Misalnya, Anda dapat menentukan bahwa pengguna hanya dapat meluncurkan instans menggunakan templat peluncuran, dan bahwa mereka hanya dapat menggunakan templat peluncuran tertentu. Anda juga dapat mengontrol parameter peluncuran yang dapat diganti pengguna di templat peluncuran. Misalnya kebijakan, lihat .Templat peluncuran

## Mengontrol penggunaan templat peluncuran

Secara default, pengguna tidak memiliki izin untuk menggunakan templat peluncuran. Anda dapat membuat kebijakan yang memberikan izin kepada pengguna untuk membuat, memodifikasi, menjelaskan, dan menghapus templat peluncuran serta versi templat peluncuran. Anda juga dapat menerapkan izin tingkat sumber daya ke beberapa tindakan templat peluncuran untuk mengontrol kemampuan pengguna untuk menggunakan sumber daya tertentu untuk tindakan tersebut. Untuk informasi selengkapnya, lihat contoh kebijakan berikut ini: Contoh: Cara menggunakan templat peluncuran.

Berhati-hatilah saat memberikan izin kepada pengguna untuk menggunakan tindakan `ec2:CreateLaunchTemplate` dan `ec2:CreateLaunchTemplateVersion`. Anda tidak dapat menggunakan izin tingkat sumber daya untuk mengontrol sumber daya mana yang dapat ditentukan pengguna dalam template peluncuran. Untuk membatasi sumber daya yang digunakan untuk meluncurkan sebuah instans, pastikan Anda memberikan izin untuk membuat templat peluncuran dan meluncurkan versi templat hanya untuk administrator yang sesuai.

## Masalah keamanan penting saat menggunakan templat peluncuran dengan Armada EC2 atau Armada Spot

> **Note**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Untuk menggunakan templat peluncuran, Anda harus memberikan izin kepada pengguna untuk membuat, memodifikasi, mendeskripsikan, dan menghapus templat peluncuran dan versi templat peluncuran. Anda dapat mengontrol siapa yang dapat membuat templat peluncuran dan meluncurkan versi templat dengan mengontrol akses ke tindakan `ec2:CreateLaunchTemplate` dan `ec2:CreateLaunchTemplateVersion`. Anda juga dapat mengontrol siapa yang dapat memodifikasi templat peluncuran dengan mengontrol akses ke tindakan `ec2:ModifyLaunchTemplate`.

> **Important**
>
> Jika Armada EC2 atau Armada Spot dikonfigurasi untuk menggunakan versi templat peluncuran Terbaru atau Default, armada tidak mengetahui apakah Terbaru atau Default yang nantinya diubah untuk menunjuk ke versi templat peluncuran yang berbeda. Ketika versi templat peluncuran yang berbeda digunakan untuk Terbaru atau Default, Amazon EC2 tidak memeriksa kembali izin untuk tindakan yang harus diselesaikan saat meluncurkan instans baru untuk memenuhi kapasitas target armada. Ini adalah pertimbangan penting saat memberikan izin kepada siapa yang dapat membuat dan mengelola versi templat peluncuran, terutama tindakan `ec2:ModifyLaunchTemplate` yang memungkinkan pengguna untuk mengubah versi templat peluncuran default.

Dengan memberikan izin kepada pengguna untuk menggunakan tindakan EC2 untuk API templat peluncuran, pengguna juga secara efektif diberi izin `iam:PassRole` jika mereka membuat atau memperbarui Armada EC2 atau Armada Spot untuk menunjuk ke versi templat peluncuran berbeda yang berisi profil instans (kontainer untuk peran IAM). Ini berarti bahwa pengguna berpotensi memperbarui template peluncuran untuk meneruskan peran IAM ke instans meskipun mereka tidak memiliki `iam:PassRole` izin. Untuk informasi selengkapnya dan contoh kebijakan IAM, lihat Menggunakan peran IAM untuk memberikan izin ke aplikasi yang berjalan di instans Amazon EC2 dalam Panduan Pengguna IAM.

Untuk informasi selengkapnya, lihat Mengontrol penggunaan templat peluncuran dan Contoh: Cara menggunakan templat peluncuran.

## Buat template peluncuran Amazon EC2

Anda dapat membuat template peluncuran Amazon EC2 dengan menentukan nilai Anda sendiri untuk parameter konfigurasi instans, atau dengan mendapatkan nilai dari template peluncuran yang ada atau instans Amazon EC2.

Anda tidak perlu menentukan nilai untuk setiap parameter dalam template peluncuran; Anda hanya perlu menentukan satu parameter konfigurasi instance untuk membuat template peluncuran. Untuk menunjukkan parameter yang Anda pilih untuk tidak ditentukan, pilih  Jangan sertakan dalam template peluncuran  saat menggunakan konsol. Saat menggunakan alat baris perintah, jangan sertakan parameter untuk menunjukkan bahwa Anda memilih untuk tidak menentukannya dalam template peluncuran.

Jika Anda ingin menentukan AMI dalam template peluncuran, Anda dapat memilih AMI, atau menentukan parameter Manajer Sistem yang akan menunjuk ke AMI pada peluncuran instance.

Saat instance diluncurkan dengan template peluncuran, nilai-nilai yang ditentukan dalam template peluncuran digunakan untuk mengonfigurasi parameter instance yang sesuai. Jika nilai tidak ditentukan dalam template peluncuran, maka nilai default untuk parameter instance yang sesuai digunakan.

Tugas

- Buat template peluncuran dengan menentukan parameter
- Buat templat peluncuran dari templat peluncuran yang ada
- Buat templat peluncuran dari instans
- Gunakan parameter Systems Manager alih-alih ID AMI

## Buat template peluncuran dengan menentukan parameter

Untuk membuat templat peluncuran, Anda harus menentukan nama templat peluncuran dan setidaknya satu parameter konfigurasi instans.

Untuk deskripsi setiap parameter, lihatReferensi untuk parameter konfigurasi EC2 instans Amazon.

Console

Untuk membuat templat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran, lalu pilih Buat templat peluncuran.
3. Di bawah Nama dan deskripsi template luncur kan, lakukan hal berikut:
   a. Untuk Nama templat peluncuran, masukkan nama deskriptif untuk templat peluncuran.
   b. Untuk Deskripsi versi templat, berikan deskripsi singkat tentang versi templat peluncuran ini.
   c. Untuk ??? menandai template peluncuran saat pembuatan, perluas tag Template, pilih Tambahkan tag baru, lalu masukkan kunci tag dan pasangan nilai. Pilih Tambahkan tanda baru lagi untuk setiap tanda tambahan yang akan ditambahkan.

> **Note**
>
> Untuk menandai sumber daya yang dibuat saat instans diluncurkan, Anda harus menentukan tanda di bawah Tag sumber daya. Untuk informasi selengkapnya, lihat Langkah 9 dalam prosedur ini.

4. Di bawah Aplikasi dan Gambar OS (Amazon Machine Image), Anda dapat memilih Jangan sertakan dalam template peluncuran, atau memilih sistem operasi (OS) untuk instance, lalu pilih AMI. Atau, Anda dapat menentukan parameter Manajer Sistem alih-alih menentukan AMI. Untuk informasi selengkapnya, lihat Gunakan parameter Systems Manager alih-alih ID AMI.

   AMI adalah template yang berisi sistem operasi dan perangkat lunak yang diperlukan untuk meluncurkan instance.

5. Di bawah Jenis instans, Anda dapat tetap memilih Jangan sertakan dalam template peluncuran, atau memilih jenis instans, atau menentukan atribut instans dan membiarkan Amazon EC2 mengidentifikasi jenis instans dengan atribut tersebut.

   > **Note**
   >
   > Menentukan atribut instans hanya didukung bila template peluncuran digunakan oleh grup Auto Scaling, EC2 Fleet, dan Spot Fleet untuk meluncurkan instans. Untuk informasi selengkapnya, lihat Membuat grup instance campuran menggunakan pemilihan jenis instance berbasis atribut dan Tentukan atribut untuk pemilihan tipe instans untuk EC2 Fleet atau Spot Fleet.
   > Jika Anda berencana untuk menggunakan template peluncuran di wizard instance peluncuran atau dengan RunInstances API, Anda tidak dapat menentukan atribut jenis instans.

   Jenis instance menentukan konfigurasi perangkat keras (CPU, memori, penyimpanan, dan kapasitas jaringan) dan ukuran komputer host yang digunakan untuk sebuah instance.

   Jika Anda tidak yakin jenis instans mana yang harus dipilih, Anda dapat melakukan hal berikut:

- Pilih Bandingkan jenis instans untuk membandingkan jenis instans yang berbeda berdasarkan atribut berikut: jumlah vCPU, arsitektur, jumlah memori (GiB), jumlah penyimpanan (GB), jenis penyimpanan, dan kinerja jaringan.
- Pilih Dapatkan saran untuk mendapatkan panduan dan saran untuk jenis instans dari pencari tipe instans EC2. Untuk informasi selengkapnya, lihat Dapatkan rekomendasi dari pencari tipe instans EC2.

> **Note**
>
> Tergantung pada saat Anda membuat akun, Anda mungkin memenuhi syarat untuk menggunakan Amazon EC2 di bawah Tingkat Gratis.
> Jika Anda membuat Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan Amazon EC2 di bawah Tingkat Gratis dengan memilih jenis instans t2.micro, atau jenis instans t 3.micro di Wilayah di mana t2.micro tidak tersedia. Ketahuilah bahwa ketika Anda meluncurkan instance t3.micro, default ke mode Unlimited, yang mungkin dikenakan biaya tambahan berdasarkan penggunaan CPU. Jika jenis instans dapat digunakan di bawah Tingkat Gratis, itu diberi label Lay ak tingkat Gratis.
> Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan jenis instans t3.micro, t3.small, t4g.micro, t4g.small, c7i-flex.large, dan m7i-flex.large selama 6 bulan atau sampai kredit Anda habis.
> Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.

6. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih Jangan sertakan dalam template peluncuran, atau pilih pasangan kunci yang ada, atau buat yang baru.
7. Di bawah Pengaturan jaringan, Anda dapat tetap memilih Jangan sertakan dalam template peluncuran, atau Anda dapat menentukan nilai untuk berbagai pengaturan jaringan.
8. Di bawah Konfigurasi penyimpanan, jika Anda menentukan AMI dalam template peluncuran, AMI menyertakan satu atau lebih volume penyimpanan, termasuk volume root (Vol ume 1 (AMI Root). Anda dapat secara opsional menentukan volume tambahan untuk dilampirkan ke instance. Untuk menambahkan volume baru, pilih Tambahkan volume baru.
9. Di bawah Tag sumber daya, untuk menandai sumber daya yang dibuat saat instance diluncurkan, pilih Tam bah tag, lalu masukkan kunci tag dan pasangan nilai. Untuk Tipe sumber daya, tentukan sumber daya yang akan ditandai pada pembuatan. Anda dapat

menentukan tanda yang sama untuk semua sumber daya, atau menentukan tanda yang berbeda untuk sumber daya yang berbeda. Pilih Tambah tanda lagi untuk setiap tanda tambahan yang akan ditambahkan.

Anda dapat menentukan tanda untuk sumber daya berikut yang dibuat saat templat peluncuran digunakan:

- Instans
- Volume
- Grafik Elastic
- Permintaan Instans Spot
- Antarmuka jaringan

> **Note**
>
> Untuk menandai templat peluncuran itu sendiri, Anda harus menentukan tag pada Tanda templat. Untuk informasi selengkapnya, lihat Langkah 3 dalam prosedur ini.

10. Untuk detail lanjutan, perluas bagian untuk melihat bidang dan secara opsional menentukan parameter tambahan apa pun untuk instance Anda.
11. Gunakan panel Ring kasan untuk meninjau konfigurasi template peluncuran Anda. Anda dapat menavigasi ke bagian mana pun dengan memilih tautannya dan kemudian membuat perubahan yang diperlukan.
12. Ketika Anda siap untuk membuat templat peluncuran Anda, pilih Buat templat peluncuran.

AWS CLI

Untuk membuat templat peluncuran

Gunakan perintah create-launch-template.

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForWebServer \
    --version-description WebVersion1 \
    --tag-specifications 'ResourceType=launch-
template,Tags=[{Key=purpose,Value=production}]' \
    --launch-template-data file://template-data.json
```

Berikut ini adalah contoh JSON yang menentukan data template peluncuran untuk konfigurasi instance. Simpan JSON ke file dan sertakan dalam `--launch-template-data` parameter seperti yang ditunjukkan pada perintah contoh.

```
{
    "NetworkInterfaces": [{
        "AssociatePublicIpAddress": true,
        "DeviceIndex": 0,
        "Ipv6AddressCount": 1,
        "SubnetId": "subnet-0abcdef1234567890"
    }],
    "ImageId": "ami-0abcdef1234567890",
    "InstanceType": "r5.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
    }],
    "CpuOptions": {
        "CoreCount":4,
        "ThreadsPerCore":2
    }
}
```

Berikut ini adalah output contoh.

```
{
    "LaunchTemplate": {
        "LatestVersionNumber": 1,
        "LaunchTemplateId": "lt-01238c059e3466abc",
        "LaunchTemplateName": "TemplateForWebServer",
        "DefaultVersionNumber": 1,
        "CreatedBy": "arn:aws:iam::123456789012:root",
        "CreateTime": "2017-11-27T09:13:24.000Z"
    }
}
```

PowerShell

Untuk membuat templat peluncuran

Gunakan New-EC2LaunchTemplate cmdlet.

```
$launchTemplateData = [Amazon.EC2.Model.RequestLaunchTemplateData]@{
    ImageId = 'ami-0abcdef1234567890'
    InstanceType = 'r5.4xlarge'
    NetworkInterfaces = @(

 [Amazon.EC2.Model.LaunchTemplateInstanceNetworkInterfaceSpecificationRequest]@{
            AssociatePublicIpAddress = $true
            DeviceIndex = 0
            Ipv6AddressCount = 1
            SubnetId = 'subnet-0abcdef1234567890'
        }
    )
    TagSpecifications = @(
        [Amazon.EC2.Model.LaunchTemplateTagSpecificationRequest]@{
            ResourceType = 'instance'
            Tags = [Amazon.EC2.Model.Tag]@{
                Key = 'Name'
                Value = 'webserver'
            }
        }
    )
    CpuOptions = [Amazon.EC2.Model.LaunchTemplateCpuOptionsRequest]@{
        CoreCount = 4
        ThreadsPerCore = 2
    }
}
$tagSpecificationData = [Amazon.EC2.Model.TagSpecification]@{
    ResourceType = 'launch-template'
    Tags = [Amazon.EC2.Model.Tag]@{
        Key = 'purpose'
        Value = 'production'
    }
}
New-EC2LaunchTemplate -LaunchTemplateName 'TemplateForWebServer' `
    -VersionDescription 'WebVersion1' `
    -LaunchTemplateData $launchTemplateData `
    -TagSpecification $tagSpecificationData
```

Berikut ini adalah output contoh.

```
CreatedBy            : arn:aws:iam::123456789012:root
```

```
CreateTime           : 9/19/2023 16:57:55
DefaultVersionNumber : 1
LatestVersionNumber  : 1
LaunchTemplateId     : lt-01238c059eEXAMPLE
LaunchTemplateName   : TemplateForWebServer
Tags                 : {purpose}
```

## Buat templat peluncuran dari templat peluncuran yang ada

Anda dapat mengklona templat peluncuran yang ada kemudian menyesuaikan parameter untuk membuat templat peluncuran baru. Namun, Anda hanya dapat melakukan ini saat menggunakan konsol Amazon EC2. AWS CLI Tidak mendukung kloning template. Untuk deskripsi setiap parameter, lihatReferensi untuk parameter konfigurasi EC2 instans Amazon.

Console

Untuk membuat templat peluncuran dari templat peluncuran yang ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran, lalu pilih Buat templat peluncuran.
3. Untuk Nama templat peluncuran, masukkan nama deskriptif untuk templat peluncuran.
4. Untuk Deskripsi versi templat, berikan deskripsi singkat tentang versi templat peluncuran ini.
5. Untuk menandai template peluncuran saat pembuatan, perluas tag Template, pilih Tambahkan tag baru, lalu masukkan kunci tag dan pasangan nilai.
6. Perluas Templat sumber, dan untuk Nama templat peluncuran, pilih templat peluncuran yang menjadi dasar templat peluncuran baru.
7. Untuk Versi templat sumber, pilih versi templat peluncuran yang menjadi dasar templat peluncuran baru.
8. Sesuaikan parameter peluncuran apa pun yang diperlukan, lalu pilih Buat templat peluncuran.

## Buat templat peluncuran dari instans

Anda dapat mengkloning parameter instans Amazon EC2 yang ada dan kemudian menyesuaikan parameter untuk membuat template peluncuran. Untuk deskripsi setiap parameter, lihatReferensi untuk parameter konfigurasi EC2 instans Amazon.

### Console

Untuk membuat templat peluncuran dari sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance, dan pilih Tind akan, Gambar dan templ at, Buat template dari instance.
4. Berikan nama, deskripsi, dan tanda, dan sesuaikan parameter peluncuran sesuai kebutuhan.

> **Note**
>
> Saat Anda membuat templat peluncuran dari sebuah instans, ID dan alamat IP antarmuka jaringan instans tersebut tidak disertakan dalam templat.

5. Pilih Buat templat peluncuran.

### AWS CLI

Anda dapat menggunakan AWS CLI untuk membuat template peluncuran dari instance yang ada dengan terlebih dahulu mendapatkan data template peluncuran dari sebuah instance, lalu membuat template peluncuran menggunakan data template peluncuran.

Untuk mendapatkan data templat peluncuran dari sebuah instans

- Gunakan perintah get-launch-template-data dan tentukan ID instans. Anda dapat menggunakan output sebagai basis untuk membuat templat peluncuran baru atau versi templat peluncuran. Secara default, output mencakup objek `LaunchTemplateData` tingkat atas, yang tidak dapat ditentukan dalam data templat peluncuran Anda. Gunakan opsi `--query` untuk mengecualikan objek ini.

```
aws ec2 get-launch-template-data \
    --instance-id i-0123d646e8048babc \
    --query "LaunchTemplateData"
```

Berikut ini adalah contoh output.

```
    {
        "Monitoring": {},
        "ImageId": "ami-8c1be5f6",
```

```
        "BlockDeviceMappings": [
            {
                "DeviceName": "/dev/xvda",
                "Ebs": {
                    "DeleteOnTermination": true
                }
            }
        ],
        "EbsOptimized": false,
        "Placement": {
            "Tenancy": "default",
            "GroupName": "",
            "AvailabilityZone": "us-east-1a"
        },
        "InstanceType": "t2.micro",
        "NetworkInterfaces": [
            {
                "Description": "",
                "NetworkInterfaceId": "eni-35306abc",
                "PrivateIpAddresses": [
                    {
                        "Primary": true,
                        "PrivateIpAddress": "10.0.0.72"
                    }
                ],
                "SubnetId": "subnet-7b16de0c",
                "Groups": [
                    "sg-7c227019"
                ],
                "Ipv6Addresses": [
                    {
                        "Ipv6Address": "2001:db8:1234:1a00::123"
                    }
                ],
                "PrivateIpAddress": "10.0.0.72"
            }
        ]
    }
```

Anda dapat menulis output langsung ke file, misalnya:

```
aws ec2 get-launch-template-data \
    --instance-id i-0123d646e8048babc \
```

```
    --query "LaunchTemplateData" >> instance-data.json
```

Untuk membuat templat peluncuran menggunakan data templat peluncuran

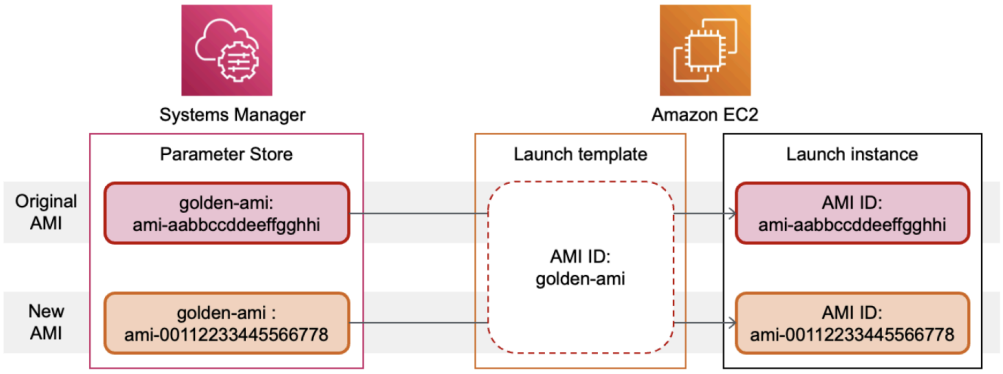
- Gunakan perintah create-launch-template untuk membuat templat peluncuran menggunakan output dari prosedur sebelumnya. Untuk informasi selengkapnya tentang membuat template peluncuran menggunakan AWS CLI, lihatBuat template peluncuran dengan menentukan parameter.

## Gunakan parameter Systems Manager alih-alih ID AMI

Alih-alih menentukan ID AMI di templat peluncuran Anda, Anda dapat menentukan parameter AWS Systems Manager . Jika ID AMI berubah, Anda dapat memperbarui ID AMI di satu tempat dengan memperbarui parameter Systems Manager di Systems Manager Parameter Store. Parameter juga dapat di bagikan dengan yang lain Akun AWS. Anda dapat menyimpan dan mengelola parameter AMI secara terpusat dalam satu akun dan membagikannya dengan setiap akun lain yang perlu merujuknya. Dengan parameter Systems Manager, semua templat peluncuran Anda dapat diperbarui dalam satu tindakan.

Parameter Manajer Sistem adalah pasangan nilai kunci yang ditentukan pengguna yang Anda buat di Penyimpanan Parameter. AWS Systems Manager Parameter Store menyediakan penyimpanan pusat untuk menyimpan nilai konfigurasi aplikasi Anda.

Dalam diagram berikut, parameter `golden-ami` terlebih dahulu dipetakan ke AMI asli `ami-aabbccddeeffgghhi` di Parameter Store. Dalam templat peluncuran, nilai untuk ID AMI adalah `golden-ami`. Saat instans diluncurkan menggunakan templat peluncuran ini, ID AMI akan diselesaikan ke `ami-aabbccddeeffgghhi`. Kemudian, AMI diperbarui menghasilkan ID AMI baru. Di Parameter Store, parameter `golden-ami` dipetakan ke `ami-00112233445566778` yang baru. Templat peluncuran tetap tidak berubah. Saat instans diluncurkan menggunakan templat peluncuran ini, ID AMI akan diselesaikan ke `ami-00112233445566778` yang baru.




Format parameter Systems Manager untuk ID AMI

Templat peluncuran mengharuskan parameter Systems Manager yang ditentukan pengguna mematuhi format berikut saat digunakan sebagai pengganti ID AMI:

- Tipe parameter: `String`
- Tipe data parameter: `aws:ec2:image` — Hal ini memastikan bahwa Parameter Store memvalidasi Anda memasukkan nilai dalam format yang tepat untuk ID AMI.

Untuk informasi selengkapnya tentang membuat parameter yang valid untuk ID AMI, lihat Membuat parameter Systems Manager di Panduan Pengguna AWS Systems Manager .

Format parameter Systems Manager dalam templat peluncuran

Untuk menggunakan parameter Systems Manager sebagai pengganti ID AMI dalam templat peluncuran, Anda harus menggunakan salah satu format berikut saat menentukan parameter dalam templat peluncuran:

Untuk mereferensikan parameter publik:

- `resolve:ssm:`*`public-parameter`*

Untuk mereferensikan parameter yang disimpan di akun yang sama:

- `resolve:ssm:`*`parameter-name`*
- `resolve:ssm:`*`parameter-name`*`:`*`version-number`* – Nomor versi itu sendiri adalah label default
- `resolve:ssm:`*`parameter-name`*`:`*`label`*

Untuk mereferensikan parameter yang dibagikan dari yang lain Akun AWS:

- `resolve:ssm:parameter-ARN`
- `resolve:ssm:parameter-ARN:version-number`
- `resolve:ssm:parameter-ARN:label`

Versi parameter

Parameter Systems Manager adalah sumber daya berversi. Saat Anda memperbarui parameter, Anda membuat versi parameter yang baru dan berurutan. Systems Manager mendukung label parameter yang dapat Anda petakan ke versi parameter tertentu.

Misalnya, parameter `golden-ami` dapat memiliki tiga versi:`1`, `2`, dan `3`. Anda dapat membuat label parameter `beta` yang memetakan ke versi `2`, dan label parameter `prod` yang memetakan ke versi `3`.

Dalam templat peluncuran, Anda dapat menentukan versi 3 parameter `golden-ami` dengan menggunakan salah satu format berikut:

- `resolve:ssm:golden-ami:3`
- `resolve:ssm:golden-ami:prod`

Menentukan versi atau label bersifat opsional. Jika versi atau label tidak ditentukan, versi terbaru parameter yang digunakan.

Tentukan parameter Systems Manager di templat peluncuran

Anda dapat menentukan parameter Systems Manager di templat peluncuran, bukan ID AMI, saat Anda membuat templat peluncuran atau templat peluncuran versi baru.

Console

Untuk menentukan parameter Systems Manager dalam templat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran, lalu pilih Buat templat peluncuran.
3. Untuk Nama templat peluncuran, masukkan nama deskriptif untuk templat peluncuran.
4. Di bawah Gambar Aplikasi dan OS (Amazon Machine Image), pilih Telusuri AMI lainnya.

5. Pilih tombol panah di sebelah kanan bilah pencarian, lalu pilih T entukan parameter value/ Systems Manajer kustom.
6. Di kotak dialog Tentukan nilai kustom atau parameter Systems Manager, lakukan hal berikut:

   a. Untuk String parameter ID AMI atau Systems Manager, masukkan nama parameter Systems Manager menggunakan salah satu format berikut:

      Untuk mereferensikan parameter publik:

      - **resolve:ssm:*public-parameter***

      Untuk mereferensikan parameter yang disimpan di akun yang sama:

      - **resolve:ssm:*parameter-name***
      - **resolve:ssm:*parameter-name*:*version-number***
      - **resolve:ssm:*parameter-name*:*label***

      Untuk mereferensikan parameter yang dibagikan dari yang lain Akun AWS:

      - **resolve:ssm:*parameter-ARN***
      - **resolve:ssm:*parameter-ARN*:*version-number***
      - **resolve:ssm:*parameter-ARN*:*label***

   b. Pilih Simpan.
7. Tentukan parameter templat peluncuran lainnya sesuai kebutuhan, lalu pilih Buat templat peluncuran.

Untuk informasi selengkapnya, lihat Buat template peluncuran dengan menentukan parameter.

AWS CLI

Untuk menentukan parameter Systems Manager dalam templat peluncuran

- Gunakan perintah create-launch-template untuk membuat templat peluncuran. Untuk menentukan AMI yang akan digunakan, masukkan nama parameter Systems Manager yang menggunakan salah satu format berikut:

  Untuk mereferensikan parameter publik:

- **resolve:ssm:*public-parameter***

Untuk mereferensikan parameter yang disimpan di akun yang sama:

- **resolve:ssm:*parameter-name***
- **resolve:ssm:*parameter-name*:*version-number***
- **resolve:ssm:*parameter-name*:*label***

Untuk mereferensikan parameter yang dibagikan dari yang lain Akun AWS:

- **resolve:ssm:*parameter-ARN***
- **resolve:ssm:*parameter-ARN*:*version-number***
- **resolve:ssm:*parameter-ARN*:*label***

Contoh berikut membuat templat peluncuran yang menetapkan hal berikut:

- Nama untuk templat peluncuran (*TemplateForWebServer*)
- Nama untuk templat peluncuran (*purpose*=*production*)
- Data untuk konfigurasi instans, yang ditentukan dalam file JSON:
  - AMI yang digunakan (`resolve:ssm:`*golden-ami*)
  - Tipe instans yang akan diluncurkan (*m5.4xlarge*)
  - Tanda untuk instans (*Name*=*webserver*)

```
aws ec2 create-launch-template \
    --launch-template-name TemplateForWebServer \
    --tag-specifications 'ResourceType=launch-
template,Tags=[{Key=purpose,Value=production}]' \
    --launch-template-data file://template-data.json
```

Berikut ini adalah contoh file JSON yang berisi data templat peluncuran untuk konfigurasi instans. Nilai untuk `ImageId` adalah nama parameter Systems Manager, yang dimasukkan dalam format `resolve:ssm:`*golden-ami* yang diperlukan.

```
{"LaunchTemplateData": {
```

```
    "ImageId": "resolve:ssm:golden-ami",
    "InstanceType": "m5.4xlarge",
    "TagSpecifications": [{
        "ResourceType": "instance",
        "Tags": [{
            "Key":"Name",
            "Value":"webserver"
        }]
    }]
  }
}
```

Verifikasi bahwa template peluncuran mendapatkan ID AMI yang benar

Untuk menyelesaikan parameter Manajer Sistem ke ID AMI yang sebenarnya

Gunakan perintah describe-launch-template-versions dan sertakan parameter. `--resolve-alias`

```
aws ec2 describe-launch-template-versions \
    --launch-template-name my-launch-template \
    --versions $Default \
    --resolve-alias
```

Tanggapan termasuk ID AMI untuk`ImageId`. Dalam contoh ini, ketika instance diluncurkan menggunakan template peluncuran ini, ID AMI diselesaikan ke`ami-0ac394d6a3example`.

```
{
    "LaunchTemplateVersions": [
        {
            "LaunchTemplateId": "lt-089c023a30example",
            "LaunchTemplateName": "my-launch-template",
            "VersionNumber": 1,
            "CreateTime": "2022-12-28T19:52:27.000Z",
            "CreatedBy": "arn:aws:iam::123456789012:user/Bob",
            "DefaultVersion": true,
            "LaunchTemplateData": {
                "ImageId": "ami-0ac394d6a3example",
                "InstanceType": "t3.micro",
            }
        }
    ]
```

```
}
```

Sumber daya terkait

Untuk informasi selengkapnya tentang bekerja dengan parameter Manajer Sistem, lihat materi referensi berikut dalam dokumentasi Manajer Sistem.

- Untuk informasi tentang cara mencari parameter publik AMI yang didukung oleh Amazon EC2, lihat Memang gil parameter publik AMI.
- Untuk informasi tentang berbagi parameter dengan AWS akun lain atau melalui AWS Organizations, lihat Bek erja dengan parameter bersama.
- Untuk informasi tentang memantau apakah parameter Anda berhasil dibuat, lihat Dukungan parameter asli untuk ID Gambar Mesin Amazon.

Batasan

- Hanya jenis Armada EC2 yang `instant` mendukung menggunakan template peluncuran yang memiliki parameter Manajer Sistem yang ditentukan sebagai pengganti ID AMI.
- Armada EC2 tipe `maintain` dan`request`, dan Armada Spot tidak mendukung penggunaan template peluncuran yang memiliki parameter Manajer Sistem yang ditentukan sebagai pengganti ID AMI. Untuk jenis Armada EC2 `maintain` dan`request`, dan untuk Armada Spot, jika Anda menentukan AMI dalam template peluncuran, Anda harus menentukan ID AMI.
- Jika Anda menggunakan pemilihan instans berbasis atribut di Armada EC2, Anda tidak dapat menentukan parameter Manajer Sistem sebagai pengganti ID AMI. Saat menggunakan pemilihan instance berbasis atribut, Anda harus menentukan ID AMI.
- Amazon EC2 Auto Scaling menyediakan batasan lain. Untuk informasi selengkapnya, lihat Menggunakan AWS Systems Manager parameter alih-alih ID AMI dalam templat peluncuran di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

## Modifikasi templat peluncuran (mengelola versi templat peluncuran)

Template peluncuran tidak dapat diubah; setelah Anda membuat templat peluncuran, Anda tidak dapat memodifikasinya. Sebagai gantinya, Anda dapat membuat templat peluncuran versi baru yang menyertakan perubahan apa pun yang Anda butuhkan.

Anda dapat membuat versi template peluncuran yang berbeda, mengatur versi default, menjelaskan versi template peluncuran, dan menghapus versi yang tidak lagi Anda butuhkan.

Tugas

- Buat versi templat peluncuran
- Menyetel versi templat peluncuran default
- Jelaskan versi templat peluncuran

## Buat versi templat peluncuran

Saat Anda membuat versi templat peluncuran, Anda dapat menentukan parameter peluncuran baru atau menggunakan versi yang sudah ada sebagai dasar untuk versi baru. Untuk deskripsi setiap parameter, lihatReferensi untuk parameter konfigurasi EC2 instans Amazon.

Console

Untuk membuat versi templat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran.
3. Pilih template peluncuran, lalu pilih T indakan, Ubah template (Buat versi baru).
4. Untuk Deskripsi versi templat, masukkan deskripsi untuk versi templat peluncuran ini.
5. (Opsional) Perluas Templat sumber dan pilih versi templat peluncuran yang akan digunakan sebagai dasar untuk versi templat peluncuran baru. Versi templat peluncuran baru mewarisi parameter peluncuran dari versi templat peluncuran ini.
6. Ubah parameter peluncuran sesuai kebutuhan.
7. Pilih Buat templat peluncuran.

AWS CLI

Untuk membuat versi templat peluncuran

Gunakan perintah create-launch-template-version. Anda dapat menentukan versi sumber yang menjadi dasar versi baru. Versi baru mewarisi parameter peluncuran dari versi ini, dan Anda dapat mengganti parameter menggunakan `--launch-template-data`. Contoh berikut membuat versi baru berdasarkan templat peluncuran versi 1 dan menentukan ID AMI yang berbeda.

```
aws ec2 create-launch-template-version \
```

```
    --launch-template-id lt-0abcd290751193123 \
    --version-description WebVersion2 \
    --source-version 1 \
    --launch-template-data "ImageId=ami-0abcdef1234567890"
```

PowerShell

Untuk membuat versi templat peluncuran

Gunakan New-EC2LaunchTemplateVersion Cmdlet. Anda dapat menentukan versi sumber yang menjadi dasar versi baru. Versi baru mewarisi parameter peluncuran dari versi ini, dan Anda dapat mengganti parameter menggunakan `LaunchTemplateData`. Contoh berikut membuat versi baru berdasarkan templat peluncuran versi 1 dan menentukan ID AMI yang berbeda.

```
New-EC2LaunchTemplateVersion `
    -LaunchTemplateId lt-0abcd290751193123 `
    -VersionDescription WebVersion2 `
    -SourceVersion 1 `
    -LaunchTemplateData (
        New-Object `
            -TypeName Amazon.EC2.Model.RequestLaunchTemplateData `
            -Property @{ImageId = 'ami-0abcdef1234567890'}
    )
```

## Menyetel versi templat peluncuran default

Anda dapat mengatur versi default untuk templat peluncuran. Saat Anda meluncurkan sebuah instans dari templat peluncuran dan tidak menentukan versinya, instans tersebut diluncurkan menggunakan parameter versi default.

Console

Untuk mengatur versi templat peluncuran default

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran.
3. Pilih template peluncuran dan pilih T indakan, Tetapkan versi default.
4. Untuk Versi templat, pilih nomor versi yang akan ditetapkan sebagai versi default dan pilih Setel sebagai versi default.

AWS CLI

Untuk mengatur versi templat peluncuran default

Gunakan perintah modify- launch-template.

```
aws ec2 modify-launch-template \
    --launch-template-id lt-0abcd290751193123 \
    --default-version 2
```

PowerShell

Untuk mengatur versi templat peluncuran default

Gunakan Edit-EC2LaunchTemplate cmdlet.

```
Edit-EC2LaunchTemplate `
    -LaunchTemplateId lt-0abcd290751193123 `
    -DefaultVersion 2
```

## Jelaskan versi templat peluncuran

Dengan menggunakan konsol, Anda dapat melihat semua versi templat peluncuran yang dipilih, atau mendapatkan daftar templat peluncuran yang versi terbaru atau default-nya cocok dengan nomor versi tertentu. Dengan menggunakan AWS CLI, Anda dapat menjelaskan semua versi, versi individual, atau berbagai versi template peluncuran tertentu. Anda juga dapat mendeskripsikan semua versi terbaru atau semua versi default dari semua templat peluncuran di akun Anda.

Console

Untuk menjelaskan versi templat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran.
3. Anda dapat melihat versi templat peluncuran tertentu, atau mendapatkan daftar templat peluncuran yang versi terbaru atau versi default-nya cocok dengan nomor versi tertentu.
   - Untuk melihat versi template peluncuran: Pilih template peluncuran. Pada tab Versi, dari Versi, pilih versi untuk melihat detailnya.

- Untuk mendapatkan daftar semua templat peluncuran yang versi terbarunya cocok dengan nomor versi tertentu: Dari bilah pencarian, pilih Versi terbaru, lalu pilih nomor versi.
- Untuk mendapatkan daftar semua templat peluncuran yang versi default-nya cocok dengan nomor versi tertentu: Dari bilah pencarian, pilih Versi default, lalu pilih nomor versi.

AWS CLI

Untuk menjelaskan versi templat peluncuran

Gunakan perintah describe-launch-template-versions dan tentukan nomor versinya. Dalam contoh berikut, versi *1* dan *3* ditentukan.

```
aws ec2 describe-launch-template-versions \
    --launch-template-id lt-0abcd290751193123 \
    --versions 1 3
```

Untuk menjelaskan versi template peluncuran terbaru dan default di akun Anda

Gunakan perintah describe-launch-template-versions dan tentukan `$Latest`, `$Default`, atau keduanya. Anda harus menghilangkan ID dan nama templat peluncuran dalam panggilan. Anda tidak dapat menentukan nomor versi.

```
aws ec2 describe-launch-template-versions \
    --versions "$Latest,$Default"
```

PowerShell

Untuk menjelaskan versi templat peluncuran

Gunakan Get-EC2TemplateVersion Cmdlet dan tentukan nomor versi. Dalam contoh berikut, versi *1* dan *3* ditentukan.

```
Get-EC2TemplateVersion `
    -LaunchTemplateId lt-0abcd290751193123 `
    -Version 1,3
```

Untuk menjelaskan versi template peluncuran terbaru dan default di akun Anda

Gunakan Get-EC2TemplateVersion Cmdlet dan tentukan`$Latest`,`$Default`, atau keduanya. Anda harus menghilangkan ID dan nama templat peluncuran dalam panggilan. Anda tidak dapat menentukan nomor versi.

```
Get-EC2TemplateVersion `
    -Version '$Latest','$Default'
```

## Menghapus template peluncuran atau versi template peluncuran

Jika Anda tidak lagi memerlukan templat peluncuran, Anda dapat menghapusnya. Menghapus templat peluncuran akan menghapus semua versinya. Jika Anda hanya ingin menghapus versi tertentu dari template peluncuran, Anda dapat melakukannya sambil mempertahankan versi lain dari template peluncuran.

Menghapus template peluncuran atau versi template peluncuran tidak memengaruhi instans apa pun yang telah Anda luncurkan dari template peluncuran.

### Hapus template peluncuran dan semua versinya

Jika Anda tidak lagi memerlukan template peluncuran, termasuk semua versinya, Anda dapat menghapus template peluncuran. Menghapus templat peluncuran akan menghapus semua versinya.

Console

Untuk menghapus template peluncuran dan semua versinya

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran.
3. Pilih template peluncuran dan pilih T indakan, Hapus template.
4. Masukkan **Delete** untuk mengonfirmasi penghapusan, lalu pilih Hapus.

AWS CLI

Untuk menghapus template peluncuran dan semua versinya

Gunakan perintah delete-launch-template dan tentukan template peluncuran.

```
aws ec2 delete-launch-template --launch-template-id lt-01238c059e3466abc
```

PowerShell

Untuk menghapus template peluncuran dan semua versinya

Gunakan perintah Remove-EC2LaunchTemplate (Alat AWS untuk PowerShell) dan tentukan template peluncuran. Jika `-Force` dihilangkan, PowerShell minta konfirmasi.

```
Remove-EC2LaunchTemplate -LaunchTemplateId lt-0123456789example -Force
```

## Hapus versi templat peluncuran

Jika Anda tidak lagi memerlukan versi templat peluncuran, Anda dapat menghapusnya.

Pertimbangan-pertimbangan

- Anda tidak dapat mengganti nomor versi setelah Anda menghapusnya.
- Anda tidak dapat menghapus template peluncuran versi default; Anda harus terlebih dahulu menetapkan versi yang berbeda sebagai default. Jika versi default adalah satu-satunya versi untuk templat peluncuran, Anda harus menghapus seluruh templat peluncuran.
- Saat menggunakan konsol, Anda dapat menghapus satu versi templat peluncuran pada satu waktu. Saat menggunakan AWS CLI, Anda dapat menghapus hingga 200 versi template peluncuran dalam satu permintaan. Untuk menghapus lebih dari 200 versi dalam satu permintaan, Anda dapat menghapus templat peluncuran, yang juga menghapus semua versinya.

Console

Untuk menghapus versi templat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Templat Peluncuran.
3. Pilih template peluncuran dan pilih T indakan, Hapus versi template.
4. Pilih versi yang akan dihapus dan pilih H apus.

AWS CLI

Untuk menghapus versi templat peluncuran

Gunakan perintah delete-launch-template-versions dan tentukan nomor versinya yang akan dihapus. Anda dapat menentukan hingga 200 versi templat peluncuran yang akan dihapus dalam satu permintaan.

```
aws ec2 delete-launch-template-versions \
    --launch-template-id lt-0abcd290751193123 \
    --versions 1
```

PowerShell

Untuk menghapus versi templat peluncuran

Gunakan Remove-EC2TemplateVersion cmdlet dan tentukan nomor versi yang akan dihapus. Anda dapat menentukan hingga 200 versi templat peluncuran yang akan dihapus dalam satu permintaan.

```
Remove-EC2TemplateVersion `
    -LaunchTemplateId lt-0abcd290751193123 `
    -Version 1
```

# Luncurkan instans Amazon EC2

Sebuah instance adalah server virtual di AWS Cloud. Anda meluncurkan instans dari Amazon Machine Image (AMI). AMI menyediakan sistem operasi, server aplikasi, dan aplikasi untuk instans Anda.

Saat Anda membuat Akun AWS, Anda dapat memulai dengan Amazon EC2 secara gratis menggunakan. AWS Tingkat Gratis Manfaat Tingkat Gratis Anda tergantung pada saat Anda membuat Akun AWS. Jika Anda membuat instans Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan Tingkat Gratis untuk meluncurkan dan menggunakan `t2.micro` instans secara gratis (di Wilayah yang tidak `t2.micro` tersedia, Anda dapat menggunakan `t3.micro` instance di bawah Tingkat Gratis). Anda dikenakan biaya untuk instans atau penggunaan yang diperhitungkan terhadap batas Tingkat Gratis Anda saat instans berjalan, meskipun instans tetap tidak aktif. Untuk informasi selengkapnya, lihat Harga Amazon EC2. Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan`t3.micro`,,`t3.small`,`t4g.micro`,`t4g.small`,`c7i-flex.large`, dan jenis `m7i-flex.large` instans selama 6 bulan atau hingga kredit Anda habis. Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.

Saat Anda meluncurkan instans, Anda dapat meluncurkan instans di subnet yang terkait dengan salah satu sumber daya berikut:

- Zona Ketersediaan — Opsi ini adalah default.
- Zona Lokal — Untuk meluncurkan instance di Zona Lokal, Anda harus memilih untuk masuk ke Zona Lokal, lalu membuat subnet di zona tersebut. Untuk informasi selengkapnya, lihat Memulai dengan Zona Lokal.
- Zona Panjang Gelombang — Untuk meluncurkan instance di Zona Panjang Gelombang, Anda harus memilih untuk masuk ke Zona Panjang Gelombang, dan kemudian membuat subnet di zona tersebut. Untuk informasi tentang cara meluncurkan instance di Zona Panjang Gelombang, lihat Memulai dengan AWS Wavelength.
- Pos Terdepan — Untuk meluncurkan instance di Pos Terdepan, Anda harus membuat Pos Terdepan. Untuk informasi tentang cara membuat Pos Terdepan, lihat Mem ulai dengan AWS Outposts.

Setelah meluncurkan instans, Anda dapat terhubung ke instans tersebut dan menggunakannya. Untuk memulai, status instans adalah `pending`. Ketika status instans adalah `running`, instans telah mulai boot. Mungkin ada waktu singkat sebelum Anda dapat terhubung ke instans. Perhatikan bahwa tipe instans bare metal mungkin membutuhkan waktu lebih lama untuk diluncurkan.

Bergantung pada bagaimana Anda berencana untuk terhubung ke instans, Anda mungkin ingin membuat konfigurasi tertentu saat meluncurkan instans. Konfigurasi ini dapat mencakup menentukan aturan grup keamanan masuk untuk lalu lintas tertentu atau mengaitkan peran profil instance. Untuk informasi lebih lanjut tentang metode koneksi yang dapat Anda gunakan untuk menghubungkan dan persyaratannya, lihatSambungkan ke instans EC2.

Instans menerima nama DNS publik yang dapat Anda gunakan untuk menghubungi instans dari internet. Instans juga menerima nama DNS privat yang dapat digunakan oleh instans lain dalam VPC yang sama untuk menghubungi instans.

Ketika Anda selesai dengan instance, untuk menghindari menimbulkan biaya yang tidak perlu, pastikan untuk menghentikannya. Untuk informasi selengkapnya, lihat Menghentikan instans Amazon EC2.

Jika Anda perlu meluncurkan instans dalam jumlah besar, menggunakan beberapa jenis instans, atau menggunakan beberapa opsi pembelian seperti Inst On-Demand ans, Instans Cadangan, dan Instans Spot, pertimbangkan untuk menggunakan EC2 Fleet. Untuk informasi selengkapnya, lihat Armada EC2 dan Armada Spot.

Jika Anda ingin mengotomatiskan siklus hidup instans, termasuk penskalaan otomatis, pemeriksaan kesehatan, dan penggantian instans yang tidak sehat, pertimbangkan untuk menggunakan Amazon EC2 Auto Scal ing.

Metode berikut adalah beberapa cara Anda dapat meluncurkan instance.

| Metode | Alat | Dokumentasi |
|---|---|---|
| Gunakan wizard instance peluncuran untuk menentukan parameter peluncuran. | Konsol Amazon EC2 | Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol |
| Buat template peluncuran dan luncurkan instance dari template peluncuran. | Konsol Amazon EC2 | Luncurkan instans EC2 menggunakan template peluncuran |
| Gunakan instance yang ada sebagai basis. | Konsol Amazon EC2 | Luncurkan instance EC2 menggunakan detail dari instance yang ada |
| Gunakan AMI yang Anda beli dari AWS Marketplace. | Konsol Amazon EC2 | Luncurkan instans Amazon EC2 dari AMI AWS Marketplace |
| Gunakan AMI yang Anda tentukan. | AWS CLI | Meluncurkan, mencantumkan, dan menghapus instans Amazon EC2 di AWS CLI |
| Gunakan AMI yang Anda tentukan. | AWS Tools for Windows PowerShell | Luncurkan Instans Amazon EC2 Menggunakan Windows PowerShell |
| Gunakan EC2 Fleet untuk menyediakan kapasitas di berbagai jenis instans EC2 dan Zona Ketersediaan, serta di seluruh opsi pembelian Inst On-Demand ans, Instans Cadangan, dan Instans Spot. | AWS CLI | Armada EC2 dan Armada Spot |

| Metode | Alat | Dokumentasi |
| --- | --- | --- |
| Gunakan CloudFormation template untuk menentukan instance. | AWS CloudFormation | AWS::EC2::Instance di Panduan Pengguna AWS CloudFormation |
| Gunakan AWS SDK khusus bahasa untuk meluncurkan instance. | AWS SDK | AWS SDK untuk.NET<br>AWS SDK untuk C++<br>AWS SDK untuk Go<br>AWS SDK untuk Java<br>AWS SDK untuk JavaScript<br>AWS SDK para PHP V3<br>AWS SDK untuk Python<br>AWS SDK para Ruby V3 |

## Tutorial untuk meluncurkan instans EC2

Ada berbagai cara untuk meluncurkan dan mengonfigurasi instans Amazon EC2. Metode dan konfigurasi yang Anda gunakan tergantung pada kasus penggunaan spesifik Anda.

Tutorial berikut dapat membantu Anda mempelajari cara meluncurkan instans EC2. Jika Anda baru mengenal Amazon EC2, kami sarankan Anda memulai dengan tutorial pertama. Tutorial dimulai dengan memperkenalkan Anda pada dasar-dasar, dan membantu Anda membangun dasar-dasarnya dengan memperkenalkan lebih banyak opsi konfigurasi.

| Tujuan | Tautan ke tutorial |
| --- | --- |
| Luncurkan instance EC2 pertama saya<br><br>Pelajari cara meluncurkan instans Amazon EC2 dengan cepat menggunakan pengaturan default di wizard instans peluncuran Amazon | Tutorial 1: Luncurkan instans Amazon EC2 pertama saya |

| Tujuan | Tautan ke tutorial |
| --- | --- |
| EC2. Pelajari juga cara meninjau bidang konfigurasi instance dan mengakhiri instance.<br><br>Durasi: 10 menit | |
| Luncurkan instance EC2 uji dan hubungkan ke sana<br><br>Pelajari cara meluncurkan instans Amazon EC2 yang dapat Anda gunakan untuk tujuan pengujian. Instans ini tidak akan memiliki konfigurasi lanjutan dan tidak akan menyimpan informasi sensitif. Anda juga akan belajar tentang pengaturan konfigurasi instans penting, cara menghubungkan ke instance, dan cara menghentikannya.<br><br>Durasi: 30 menit | Tutorial 2: Luncurkan instance EC2 uji dan hubungkan ke sana |

## Mencari tutorial lain?

- Tutorial: Instal server LAMP di AL2023
- Tutorial: Konfigurasikan SSL/TLS pada AL2023
- Tutorial: Selenggarakan WordPress blog di AL2023
- Tutorial: Selesaikan konfigurasi yang diperlukan untuk menyambung ke instans Anda menggunakan EC2 Instance Connect
- Tutorial: Hubungkan instans Amazon EC2 ke basis data Amazon RDS

## Tutorial 1: Luncurkan instans Amazon EC2 pertama saya

| Atribut | Nilai |
| --- | --- |
| Tujuan tutorial | Pelajari cara meluncurkan instans Amazon EC2 dengan cepat menggunakan pengaturan |

| Atribut | Nilai |
| --- | --- |
| | default di wizard instans peluncuran Amazon EC2. Pelajari juga cara meninjau bidang konfigurasi instance dan mengakhiri instance. |
| Pengalaman EC2 | Pemula |
| Durasi | 10 menit |

| Atribut | Nilai |
| --- | --- |
| Biaya | Tingkat Gratis memenuhi syarat<br><br>Saat Anda membuat Akun AWS, Anda dapat memulai dengan Amazon EC2 secara gratis menggunakan. AWS Tingkat Gratis<br><br>Jika Anda membuat Akun AWS sebelum 15 Juli 2025, usianya kurang dari 12 bulan, dan Anda belum melebihi manfaat Tingkat Gratis untuk Amazon EC2, Anda tidak akan dikenakan biaya apa pun untuk menyelesaikan tutorial ini, karena kami membantu Anda memilih opsi yang ada dalam manfaat Tingkat Gratis. Jika tidak, Anda akan dikenakan biaya penggunaan Amazon EC2 standar sejak Anda meluncurkan instans (meskipun instans tetap idle) hingga Anda menghentikannya.<br><br>Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, usianya kurang dari 6 bulan, dan Anda belum menghabiskan semua kredit Anda, Anda tidak akan dikenakan biaya apa pun untuk menyelesaikan tutorial ini, karena kami membantu Anda memilih opsi yang ada dalam manfaat Tingkat Gratis.<br><br>Untuk informasi tentang cara menentukan apakah Anda memenuhi syarat untuk Tingkat Gratis, lihatthe section called “Melacak penggunaan Tingkat Gratis Anda”. |

| Atribut | Nilai |
| --- | --- |
| Prasyarat | • Anda harus memiliki AWS akun, mengonfig urasi pengguna dengan akses administrator, dan menggunakan pengguna administrator untuk masuk ke Konsol Manajemen AWS. Tidak yakin bagaimana melakukan ini? Coba tutorial ini: Menyiapkan AWS Lingkungan Anda<br>• Anda harus memiliki keakraban umum dengan AWS konsol. Tidak yakin harus memulai dari mana? Coba panduan memulai ini: Mem ulai dengan Konsol Manajemen AWS |

Gambaran umum tutorial

Tutorial ini dirancang untuk pemula tanpa pengalaman sebelumnya menggunakan Amazon EC2. Kami akan memandu Anda melalui langkah-langkah untuk membuat—kami menyebutnya pel uncuran —instans EC2 pertama Anda menggunakan konsol EC2. Sebuah instance pada dasarnya adalah server web di AWS Cloud. Setelah meluncurkan instance Anda, kami akan menunjukkan cara menemukannya di konsol. Terakhir, untuk membantu Anda mengelola biaya, kami akan menunjukkan cara menghapus—kami menyebutnya meng akhir—instance Anda.

Tutorial ini dibagi menjadi tugas-tugas singkat berikut. Anda harus menyelesaikan setiap tugas sebelum pindah ke yang berikutnya.

- Tugas 1: Luncurkan instance Anda
- Tugas 2: Temukan instance Anda
- Tugas 3: Melihat konfigurasi instans Anda
- Tugas 4: Hentikan instance Anda

Tugas 1: Luncurkan instance Anda

Dalam tugas ini, Anda akan mengambil jalur tercepat untuk meluncurkan instance Anda dengan hanya melakukan hal-hal penting. Kami akan menggunakan wizard instance peluncuran EC2,

formulir berbasis web yang menyediakan semua bidang untuk mengonfigurasi dan meluncurkan instans Anda. Ini menyederhanakan proses dengan memberikan nilai default untuk bidang konfigurasi instance.

Sebelum Anda mulai

Pastikan Anda telah menyelesaikan prasyarat yang tercantum dalam tabel sebelumnya, termasuk masuk Konsol Manajemen AWS dengan pengguna administrator Anda.

Ikuti langkah-langkah ini untuk meluncurkan instans Anda dengan cepat

1. Buka konsol Amazon EC2:

   Kunjungi https://console.aws.amazon.com/ec2/.

2. Buka wizard instance peluncuran EC2:

   Dari dasbor EC2, pilih  Luncurkan instance.

   Formu  lir Luncurkan instance  berbasis web terbuka. Ini adalah wizard instance peluncuran EC2.

3. Beri nama instans Anda:

   Di bawah  Nama dan tag, untuk  Nama, masukkan nama deskriptif seperti**`My first EC2 instance`**.

   Meskipun penamaan instans Anda tidak diperlukan, ini membantu mengidentifikasi instance Anda nanti.

4. Lanjutkan tanpa pasangan kunci:

   Di bawah Pasangan  kunci (login), untuk Nama pasangan  kunci, pilih L  anjutkan tanpa pasangan kunci (Tidak disarankan).

   Pasangan kunci dapat digunakan untuk login aman. Namun, karena kami tidak akan masuk ke instance dalam tutorial ini, Anda tidak memerlukan pasangan kunci untuk saat ini.

5. Luncurkan instance Anda:

   Di panel Ringkasan di sebelah kanan, pilih Luncurkan instans.

   Amazon EC2 meluncurkan instans Anda dengan cepat menggunakan pengaturan default. Span duk  Sukses mengkonfirmasi peluncuran.

Selamat! Anda telah berhasil meluncurkan instans EC2 pertama Anda!

Tugas 2: Temukan instance Anda

Dalam tugas ini, Anda akan menemukan instance yang baru saja Anda luncurkan di konsol EC2.

Ikuti langkah-langkah berikut untuk menemukan instans Anda di konsol EC2

1. Buka halaman Instans:

   Jika Anda masih berada di halaman sukses, pilih Inst ans di remah roti di bagian atas layar. Anda mungkin perlu memilih tiga elips terlebih dahulu untuk mengaksesnya.

   Jika Anda telah menavigasi jauh, pilih Instans dari panel navigasi.
2. Temukan instans Anda:

   Di kolom Nama, temukan instance Anda dengan nama yang Anda berikan.

Tugas 3: Melihat konfigurasi instans Anda

Dalam tugas ini, Anda akan terbiasa melihat detail konfigurasi instans Anda.

Ikuti langkah-langkah berikut untuk melihat konfigurasi instans Anda

1. Cari ID instance:

   Di kolom ID Instance, catat ID unik instans Anda. Dimulai dengan i— diikuti oleh 17 karakter alfanumerik, misalnya, i- 01aeed690c9fb5322.

   ID instans secara otomatis ditetapkan ke instans Anda saat diluncurkan.
2. Buka halaman detail instance:

   Di kolom ID Instance, pilih tautan ID untuk membuka halaman detail instance tempat Anda dapat meninjau konfigurasinya.
3. Jelajahi detail konfigurasi instance:

   Luangkan beberapa menit untuk menjelajahi detail konfigurasi instans Anda. Dalam tutorial berikutnya, kita akan menyelami lebih dalam konfigurasi. Untuk saat ini, gunakan waktu ini untuk membiasakan diri dengan halaman detail instance.

Tip: Untuk menemukan bidang dengan cepat, tekan Ctrl+F atau Command+F pada keyboard Anda.

a. Jenis instans: Dapatkah Anda menemukan jenis instans? Mungkin t3.micro, misalnya.

b. Alamat IPv4 publik: Dapatkah Anda menemukan alamat IPv4 publik yang dialokasikan ke instans Anda? Ini dalam format yang mirip dengan contoh berikut: 34.24 2.148.128.

c. Pemilik instans: Dapatkah Anda mengidentifikasi pemilik instance ini? Ini kamu! Akun AWS Nomor Anda tercantum di bawah bidang Pemilik.

d. Tag instance: Nama yang Anda berikan pada instance Anda sebenarnya adalah tag. Dapatkah Anda menemukan tag instance Anda? Pilih tab Tanda. Kuncinya adalah Nama, dan nilainya adalah nama yang Anda berikan.

e. Waktu peluncuran: Dapatkah Anda menemukan kapan Anda meluncurkan instans Anda? Pilih tab Detail dan temukan bidang Waktu peluncuran.

f. Status instans: Dapatkah Anda memverifikasi status instance Anda? Seharusnya berjalan.

Luangkan beberapa menit lagi untuk menjelajahi bidang konfigurasi instance lainnya. Ketika Anda siap, lanjutkan ke tugas berikutnya.

Tugas 4: Hentikan instance Anda

> **Warning**
>
> Mengakhiri instance bersifat permanen dan tidak dapat diubah.
> Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
> Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Dalam tugas ini, Anda akan menghapus instans untuk mempertahankan manfaat Tingkat Gratis Anda. Di EC2, termin asi adalah istilah yang digunakan untuk menghapus instance.

Ikuti langkah-langkah berikut untuk mengakhiri instance Anda

1. Memulai penghentian:

   Jika Anda masih berada di halaman detail instans, pilih menu Status instans (kanan atas), lalu pilih H entikan (hapus) instance.

   Jika Anda telah menavigasi jauh, pilih Instans dari panel navigasi. Kemudian, pada halaman Instans, pilih kotak centang di sebelah nama instance Anda, lalu pilih menu Status Instance (kanan atas), lalu pilih Hentikan (hapus) instance.
2. Konfirmasikan penghentian:

   Di jendela Hentikan (hapus) instance yang terbuka, pilih tombol Hentikan (hapus) untuk mengonfirmasi bahwa Anda ingin mengakhiri instance Anda.
3. Memantau status instans:

   Pada halaman Instans, periksa kolom Status Instance. Status instans Anda berubah menjadi Shutting-down. Jika Anda tidak melihat teks lengkap, coba perlebar kolom.

   Setelah instans dimatikan, Amazon EC2 menghapus instans, dan instans menghilang dari halaman Inst ans.

Kuncinya yang menarik

Dalam tutorial ini, Anda membahas konsep-konsep kunci berikut:

- Inst ance mengacu pada server web Amazon EC2 di AWS Cloud.
- Peluncuran mengacu pada pembuatan instance EC2.
- Menghentikan mengacu pada penghapusan instance EC2.
- Wizard instance peluncuran EC2 berisi nilai default untuk konfigurasi instance, memungkinkan peluncuran instans dengan cepat dan mudah.
- ID instance adalah pengidentifikasi unik yang secara otomatis ditetapkan ke instance Anda, sedangkan nama instans adalah tag opsional yang dapat Anda tetapkan untuk memudahkan identifikasi.

Langkah selanjutnya

Untuk membangun kepercayaan dalam meluncurkan dan mengakhiri instance, pertimbangkan untuk mengulangi langkah-langkah dalam tutorial ini. Pastikan untuk menghentikan instans apa pun yang Anda luncurkan untuk mempertahankan manfaat Tingkat Gratis Anda.

Setelah Anda merasa nyaman dengan dasar-dasar ini, lanjutkan ke tutorial berikutnya, yang memberikan penyelaman lebih dalam ke bidang konfigurasi instance utama.

## Tutorial 2: Luncurkan instance EC2 uji dan hubungkan ke sana

| Atribut | Nilai |
| --- | --- |
| Tujuan tutorial | Pelajari cara meluncurkan instans Amazon EC2 yang dapat Anda gunakan untuk tujuan pengujian. Instans ini tidak akan memiliki konfigurasi lanjutan dan tidak akan menyimpan informasi sensitif. Anda juga akan belajar tentang pengaturan konfigurasi instans penting, cara menghubungkan ke instance, dan cara menghentikannya. |
| Pengalaman EC2 | Pemula |
| Durasi | 30 menit |
| Biaya | Tingkat Gratis memenuhi syarat<br><br>Saat Anda membuat Akun AWS, Anda dapat memulai dengan Amazon EC2 secara gratis menggunakan. AWS Tingkat Gratis<br><br>Jika Anda membuat Akun AWS sebelum 15 Juli 2025, usianya kurang dari 12 bulan, dan Anda belum melebihi manfaat Tingkat Gratis untuk Amazon EC2, Anda tidak akan dikenakan biaya apa pun untuk menyelesaikan tutorial ini, karena kami membantu Anda memilih opsi yang ada dalam manfaat Tingkat Gratis. Jika |

| Atribut | Nilai |
| --- | --- |
| | tidak, Anda akan dikenakan biaya penggunaan Amazon EC2 standar sejak Anda meluncurkan instans (meskipun instans tetap idle) hingga Anda menghentikannya.<br><br>Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, usianya kurang dari 6 bulan, dan Anda belum menghabiskan semua kredit Anda, Anda tidak akan dikenakan biaya apa pun untuk menyelesaikan tutorial ini, karena kami membantu Anda memilih opsi yang ada dalam manfaat Tingkat Gratis.<br><br>Untuk informasi tentang cara menentukan apakah Anda memenuhi syarat untuk Tingkat Gratis, lihatthe section called “Melacak penggunaan Tingkat Gratis Anda”. |
| Prasyarat | Selesaikan Tutorial 1: Luncurkan instans Amazon EC2 pertama saya. |

Gambaran umum tutorial

Tutorial ini dirancang untuk pemula yang ingin meluncurkan instance EC2 yang dapat mereka gunakan untuk tujuan pengujian.

Kami akan menjelaskan bidang konfigurasi instans utama, dan kemudian memandu Anda melalui langkah-langkah untuk meluncurkan instance uji menggunakan nilai default di konsol EC2. Setelah meluncurkan instans Anda, kami akan menunjukkan cara masuk — kami menyebutnya connect to — instance Anda. Kami juga akan menunjukkan cara membuat pasangan kunci, yang diperlukan untuk menghubungkan ke instance Anda dalam tutorial ini. Terakhir, untuk membantu mengelola biaya, kami akan menunjukkan cara menghentikan instans Anda untuk menghindari biaya penggunaan.

Anda akan meluncurkan instance Linux dalam tutorial ini. Sementara langkah-langkah dalam tutorial ini dapat digunakan untuk meluncurkan instance dengan sistem operasi lain, instruksi untuk menghubungkan ke instance khusus untuk instance Linux.

Tutorial ini dibagi menjadi tugas-tugas singkat berikut. Anda harus menyelesaikan setiap tugas sebelum pindah ke yang berikutnya.


- Tugas 1: Biasakan diri Anda dengan komponen kunci untuk meluncurkan instance
- Tugas 2: Tinjau diagram teknis
- Tugas 3: Buat pasangan kunci
- Tugas 4: Luncurkan instance pengujian Anda
- Tugas 5: Temukan instance Anda
- Tugas 6: Melihat konfigurasi instans
- Tugas 7: Biasakan diri Anda dengan komponen kunci untuk menghubungkan ke instance
- Tugas 8: Hubungkan ke instance Anda
- Tugas 9: Hentikan instance Anda


## Tugas 1: Biasakan diri Anda dengan komponen kunci untuk meluncurkan instance

Dalam tugas ini, Anda akan menjelajahi komponen kunci yang diperlukan untuk meluncurkan instance EC2. Ini adalah AMI, tipe instance, pasangan kunci, grup keamanan, jaringan (VPC dan subnet), dan volume Amazon EBS. Anda juga akan menjelajahi komponen opsional, tag Nama.

Untuk membantu memvisualisasikan komponen-komponen ini, pikirkan contoh seperti rumah sewaan. Sama seperti menyewa rumah memberi Anda tempat tinggal tanpa perlu memiliki dan memelihara properti, instans EC2 menyediakan daya komputasi tanpa Anda perlu memiliki dan memelihara infrastruktur yang mendasarinya.

Saat memutuskan jenis instance yang akan diluncurkan, Anda akan mempertimbangkan kriteria konfigurasi untuk instance tersebut, sama seperti Anda akan mempertimbangkan kriteria yang Anda inginkan dari sebuah rumah. Meskipun analogi ini menyederhanakan berbagai hal, ini menawarkan cara yang bermanfaat untuk memvisualisasikan komponen sampai Anda lebih akrab dengannya.

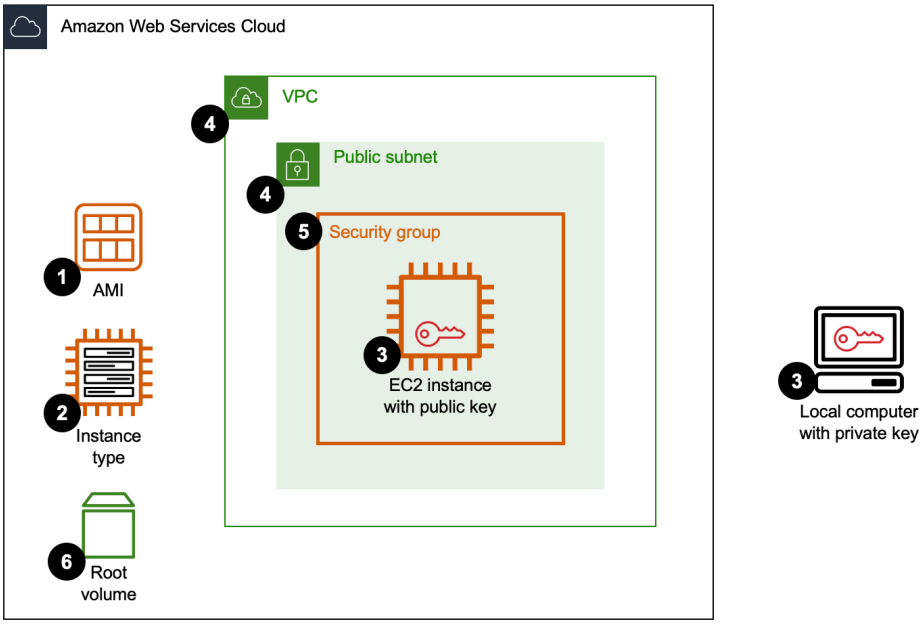
- AMI — Bahan dan fasilitas bangunan rumah: Amazon Machine Image (AMI) menentukan sistem operasi dan aplikasi yang digunakan instans Anda untuk memulai. Ini seperti memilih bahan bangunan (seperti batu bata, baja, atau kayu) dan fasilitas (seperti peralatan dan perabotan) rumah

Anda. AMI dasar seperti rumah tanpa perabotan dengan peralatan dasar, sedangkan AMI khusus dengan perangkat lunak pra-instal seperti rumah berperabotan lengkap.

- Jenis instans — Ukuran rumah dan daya: Jenis instans menentukan ukuran dan kemampuan instans EC2 Anda, seperti memilih ukuran rumah, jumlah kamar, dan kapasitas energi. Setiap jenis instans menentukan jumlah CPU, memori, penyimpanan, dan kapasitas jaringan instans Anda. AMI yang dipilih mungkin membatasi jenis instans apa yang dapat Anda pilih.
- Pasangan kunci — Kunci pintu depan: Pasangan kunci seperti kunci dan kunci pintu depan rumah Anda. Kunci publik bertindak sebagai kunci pada instance Anda, sedangkan kunci pribadi adalah kunci yang harus Anda simpan dengan aman di komputer lokal Anda. Jika orang lain memegang kunci pribadi Anda, mereka dapat mengakses instance Anda, seperti bagaimana seseorang dengan kunci pintu depan Anda dapat memasuki rumah Anda.
- Jaringan (VPC dan subnet) - Batas properti, area yang dibagian, dan nomor rumah: Cloud pribadi virtual Anda (VPC) seperti seluruh properti tempat rumah Anda berada, dan subnet adalah area yang dipisahkan di sekitar rumah. Jika Anda memiliki beberapa rumah (instance) di properti Anda, Anda mungkin ingin membaginya menjadi area yang berbeda (subnet yang berbeda) tergantung pada tujuannya. Beberapa rumah memungkinkan pengunjung untuk berkeliaran bebas melalui taman (subnet publik dengan akses internet), sementara yang lain memiliki taman berpagar untuk membatasi masuk (subnet pribadi tanpa akses internet). Setiap subnet berisi berbagai alamat IP, seperti nomor rumah, yang dapat ditetapkan ke instance dalam subnet.
- Kelompok keamanan — Penjaga gerbang: Kelompok keamanan bertindak seperti penjaga gerbang, mengendalikan siapa yang diizinkan mengunjungi rumah Anda. Ini memberlakukan seperangkat aturan yang mengontrol lalu lintas apa yang diizinkan untuk mencapai instance Anda. Misalnya, aturan yang memungkinkan lalu lintas SSH dari alamat IP tertentu seperti gatekeeper yang mengizinkan hanya orang tertentu untuk mengirimkan bahan makanan. Demikian pula, mengizinkan lalu lintas HTTPS dari mana saja seperti membiarkan publik datang dan melihat bagian luar rumah Anda.
- Volume Amazon EBS — Unit penyimpanan: Volume EBS seperti unit penyimpanan tempat Anda dapat menyimpan barang-barang Anda. Setiap instance memiliki volume root (tempat AMI disimpan), dan Anda dapat menambahkan lebih banyak volume (penyimpanan) kapan saja sesuai kebutuhan.
- Tag nama — Nama rumah: Tag nama berfungsi seperti tanda di rumah, membantu Anda dengan mudah mengidentifikasi siapa yang tinggal di sana. Meskipun tag Name memudahkan untuk membedakan antara instance, tag itu tidak diperlukan saat meluncurkan instance.

## Tugas 2: Tinjau diagram teknis

Dalam tugas ini, Anda akan terbiasa dengan diagram teknis khas yang kami gunakan dalam AWS dokumentasi. Diagram berikut mewakili konfigurasi untuk instance pengujian yang akan Anda luncurkan dalam tutorial ini. Pada tugas sebelumnya, kami memperkenalkan komponen-komponen ini menggunakan analogi rumah sewa. Sekarang, kita akan fokus pada komponen EC2 yang sebenarnya. Label bernomor sesuai dengan deskripsi berikut.




1. AMI — AMI adalah gambar yang Anda pilih saat meluncurkan instance. Ini adalah template yang berisi sistem operasi dan perangkat lunak untuk dijalankan pada instance Anda. Misalnya, jika Anda ingin meluncurkan instance Linux, Anda dapat memilih Amazon Linux 2023 AMI. Atau, jika Anda ingin meluncurkan instance Windows, Anda dapat memilih AMI Dasar Microsoft Windows Server 2022. Katalog AMI di konsol Amazon EC2 berisi ribuan gambar untuk dipilih.
2. Jenis instans — Jenis instance adalah perangkat keras yang menentukan CPU, memori, penyimpanan, dan kapasitas jaringan komputer host yang digunakan untuk instance Anda. Amazon EC2 menawarkan lebih dari 600 jenis instans untuk dipilih, masing-masing bervariasi dalam konfigurasi dan ukuran perangkat keras, memungkinkan Anda memilih yang paling sesuai dengan kebutuhan aplikasi Anda.
3. Pasangan kunci — Pasangan kunci adalah kumpulan kredentif keamanan yang Anda gunakan untuk membuktikan identitas Anda saat menghubungkan ke instans Anda. Kunci publik ada di instance Anda dan kunci pribadi ada di komputer lokal Anda.

Di EC2, menghubungkan ke instance Anda mengacu pada masuk ke instans Anda dari komputer lokal Anda. Meskipun ada cara lain untuk terhubung dengan aman ke instance Anda, dalam tutorial ini kami menggunakan pasangan kunci.

4. Jaringan — Jaringan terdiri dari VPC dan satu atau lebih subnet. VPC adalah jaringan virtual di dalam AWS Cloud. Setiap AWS pelanggan memiliki VPC mereka sendiri yang didedikasikan untuk mereka Akun AWS. Anda akan meluncurkan instance Anda ke subnet di VPC Anda. Subnet adalah rentang alamat IP dalam VPC. Subnet default Anda adalah subnet publik, yang berarti akan menetapkan alamat IP publik dan menyediakan akses internet ke instance Anda dari luar jaringan Amazon.
5. Grup keamanan — Grup keamanan bertindak sebagai firewall untuk mengontrol lalu lintas ke instans Anda. Grup keamanan berisi aturan yang memungkinkan jenis lalu lintas tertentu masuk ke instans Anda. Untuk terhubung melalui SSH dari komputer lokal Anda ke instance Anda (menggunakan pasangan kunci Anda), Anda memerlukan aturan yang memungkinkan lalu lintas SSH dari komputer lokal Anda.
6. Volume EBS — Volume Amazon EBS adalah perangkat penyimpanan yang berfungsi seperti hard drive fisik. Instans Anda dilengkapi dengan volume root, yang merupakan volume EBS khusus yang menyimpan AMI dengan sistem operasi dan perangkat lunak yang diperlukan untuk mem-boot instans Anda. Anda dapat menambahkan volume data secara opsional. Namun, karena instance pengujian Anda tidak akan menyimpan data sensitif apa pun, Anda tidak memerlukan volume data terenkripsi tambahan.

Selamat! Anda telah menyelesaikan tugas konseptual dalam tutorial ini. Dalam tugas berikut, Anda akan menggunakan konsol Amazon EC2 untuk membuat komponen yang telah Anda pelajari.

## Tugas 3: Buat pasangan kunci

Dalam tugas ini, Anda akan membuat pasangan kunci. Pasangan kunci terdiri dari dua bagian: kunci publik, yang akan Anda tambahkan ke instans Anda, dan kunci pribadi yang cocok, yang akan Anda gunakan untuk terhubung dengan aman ke instans Anda. Pada tugas berikutnya, Anda akan memilih pasangan kunci ini saat meluncurkan instance Anda, yang secara otomatis menambahkan kunci publik ke instance. Sangat penting untuk menyimpan kunci pribadi dengan aman di komputer lokal Anda, karena siapa pun yang memiliki akses ke kunci tersebut dapat terhubung ke instance Anda.

Jika Anda lebih suka menggunakan pasangan kunci yang ada saat meluncurkan instance pengujian, jangan ragu untuk melewatkan tugas ini. Jika tidak, lanjutkan untuk membuat pasangan kunci baru.

### Sebelum Anda mulai

Pastikan Anda telah menyelesaikan prasyarat yang tercantum dalam tabel sebelumnya, termasuk masuk Konsol Manajemen AWS dengan pengguna administrator Anda.

Ikuti langkah-langkah ini untuk membuat pasangan kunci

1. Buka konsol Amazon EC2:

   Kunjungi https://console.aws.amazon.com/ec2/.

2. Arahkan ke halaman konsol Pasangan kunci:

   Pada panel navigasi, di Jaringan & Keamanan, pilih Pasangan Kunci.

   - Jika sebelumnya Anda membuat pasangan kunci, mereka muncul di tabel.
   - Jika tidak ada pasangan kunci, tabel kosong.

3. Buat pasangan kunci baru:

   Pilih tombol Buat pasangan kunci (kanan atas) untuk membuka formulir Buat pasangan kunci berbasis web, dan masukkan detail pasangan kunci Anda, sebagai berikut:

   a. Beri nama pasangan kunci Anda: Untuk Nama, masukkan nama yang akan membantu Anda mengenali pasangan kunci, seperti**`test-instance-key-pair`**.

      Nama bisa maksimal 255 karakter ASCII. Tidak boleh mengandung spasi di depan maupun belakang.

   b. Pilih jenis pasangan kunci: Untuk jenis pasangan kunci, pilih ED25519.

      Instans Linux mendukung tipe kunci RSA dan ED25519, sementara instans Windows hanya mendukung RSA. Karena Anda akan meluncurkan instance Linux dalam tutorial ini, Anda dapat menggunakan kunci ED25519.

   c. Pilih format file kunci pribadi: Untuk format file kunci pribadi, pilih .pem.

      Ini adalah format di mana file kunci pribadi Anda akan disimpan.

4. Simpan kunci publik ke Amazon EC2 dan unduh kunci pribadi:

   Pilih tombol Buat pasangan kunci (kanan bawah).

   Amazon EC2 menyimpan kunci publik, sementara browser Anda mengunduh file kunci pribadi secara otomatis ke komputer lokal Anda. File diberi nama sesuai dengan nama yang Anda

tentukan untuk pasangan kunci, dan ekstensi adalah format file yang Anda pilih. Pindahkan file kunci pribadi ke lokasi aman di komputer Anda.

> **Important**
>
> Ini adalah satu-satunya kesempatan Anda harus menyimpan file kunci pribadi.

5. Atur izin pada kunci (untuk pengguna macOS dan Linux):

   Jika Anda berencana untuk menyambung ke instance menggunakan SSH di komputer macOS atau Linux, Anda harus menetapkan izin yang benar untuk file kunci pribadi Anda. Buka jendela terminal dan jalankan perintah berikut, ganti *test-instance-key-pair* dengan nama pasangan kunci Anda:

   ```
   chmod 400 test-instance-key-pair.pem
   ```

   Perintah ini memastikan bahwa hanya Anda yang dapat membaca file kunci pribadi, yang diperlukan untuk membuat koneksi aman ke instance Anda. Tanpa izin ini, Anda tidak akan dapat terhubung menggunakan pasangan kunci ini.

Selamat! Anda telah berhasil membuat pasangan kunci!

Tugas 4: Luncurkan instance pengujian Anda

Dalam tugas ini, Anda akan segera meluncurkan instance uji menggunakan wizard instance peluncuran EC2. Anda akan mengonfigurasi pengaturan konfigurasi instance utama untuk instance Linux dan menggunakan nilai default untuk bidang lainnya.

Untuk membantu mengelola biaya, sebaiknya pilih komponen yang memenuhi syarat tingkat gratis.

Ikuti langkah-langkah ini untuk meluncurkan instance pengujian

1. Buka konsol Amazon EC2:

   Kunjungi https://console.aws.amazon.com/ec2/.

2. Buka wizard instance peluncuran EC2:

   Dari dasbor EC2, pilih Luncurkan instance.

   Formu lir Luncurkan instance berbasis web terbuka. Ini adalah wizard instance peluncuran EC2.

3. Beri nama instans Anda:

   Di bawah Nama dan tag, untuk Nama, masukkan nama deskriptif seperti**`Test instance`**.

   Nama instans adalah tanda, di mana kuncinya adalah Name, dan nilainya adalah nama yang Anda tentukan.

   Tip: Untuk contoh pengujian, tag nama sudah cukup. Namun, untuk instans produksi, praktik terbaik adalah menetapkan kebijakan penandaan untuk menstandarkan penandaan di semua sumber daya Anda.

4. Pilih sistem operasi dan perangkat lunak—Amazon Machine Image (AMI):

   Di bawah Gambar Aplikasi dan OS (Amazon Machine Image), untuk Amazon Machine Image (AMI), pilihan defaultnya adalah Amazon Linux 2023 AMI. AMI ini ditandai dengan memenuhi syarat tingkat gratis. Dalam tutorial ini, Anda akan meluncurkan instance Linux, jadi biarkan pengaturan default tetap di bawah batas Tingkat Gratis.

5. Pilih perangkat keras Anda—jenis instans:

   Di bawah Jenis Inst ance, untuk Jenis Inst ance, pertahankan pilihan default untuk tutorial ini. Jenis instans default dapat digunakan di bawah Tingkat Gratis dan perangkat kerasnya cocok untuk instance pengujian Anda.

6. Bersiaplah untuk login aman dengan pasangan kunci:

   Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang Anda buat di tugas sebelumnya. Jika Anda tidak melihat pasangan kunci Anda dalam daftar, pilih ikon refresh (di sebelah kanan daftar).

   Saat instance Anda diluncurkan, itu akan menempatkan kunci publik pada instance. Untuk menyambung ke instance setelah diluncurkan, Anda akan menggunakan kunci pribadi yang sesuai yang Anda unduh di tugas sebelumnya.

7. Konfigurasikan pengaturan jaringan untuk mengaktifkan akses internet:

   Di bawah Pengaturan jaringan, bidang Jaringan (VPC Anda) dan Subnet dikonfigurasi secara default. Pertahankan pengaturan default untuk tutorial ini untuk membantu Anda memulai dengan cepat. Jika Anda belum mengubah subnet default, instans Anda akan memiliki akses internet.

   Tip: Subnet default Anda adalah subnet publik, yang berarti subnet akan menetapkan alamat IP publik dan menyediakan akses internet ke instans Anda dari luar jaringan Amazon. Untuk

contoh pengujian, tidak apa-apa menggunakan pengaturan subnet default yang menyediakan akses internet. Namun, untuk contoh produksi, praktik terbaik adalah hanya menetapkan alamat IP publik dan menggunakan subnet dengan akses internet bila benar-benar diperlukan.

8. Siapkan firewall instance (grup keamanan):

   Di bawah  Pengaturan jaringan, di bawah  Firewall (grup keamanan), tetap pilih kotak centang Iz  inkan lalu lintas SSH dari mana saja (0.0.0.0). Ini akan membuat grup keamanan baru untuk instance pengujian Anda yang memungkinkan lalu lintas SSH dari alamat IP apa pun.

   Grup keamanan bertindak sebagai firewall untuk mengontrol lalu lintas ke instance Anda. Untuk terhubung melalui SSH dari komputer lokal Anda ke instance Anda, Anda memerlukan aturan yang memungkinkan lalu lintas SSH dari komputer lokal Anda.

   Tip:  Alamat IP komputer lokal Anda mungkin berubah seiring waktu jika penyedia layanan internet Anda menggunakan penetapan IP dinamis. Kami berasumsi bahwa ketika Anda menggunakan instance untuk tujuan pengujian, Anda tidak akan menggunakan instance untuk menyimpan informasi sensitif, dan oleh karena itu langkah-langkah keamanan bisa kurang ketat. Untuk contoh pengujian, umumnya dapat diterima untuk mengizinkan lalu lintas dari alamat IP apa pun (`0.0.0.0/0`) sehingga Anda selalu dapat terhubung meskipun alamat IP Anda berubah. Namun, untuk instans produksi, terutama yang memiliki data sensitif, praktik terbaik adalah mengizinkan lalu lintas hanya dari alamat IP yang diketahui.

9. Konfigurasikan penyimpanan instance:

   Di bawah  Konfigurasi penyimpanan,  bidang  Root volume (Encrypted) dikonfigurasi secara default. Biarkan pengaturan sebagaimana adanya agar tetap berada di bawah batas Tingkat Gratis.

   Karena instance pengujian kami tidak akan menyimpan data sensitif apa pun, kami tidak memerlukan volume data terenkripsi tambahan.

10. Tinjau konfigurasi instance:

    Di  panel  Ringkasan di sebelah kanan, Anda dapat meninjau pengaturan tingkat tinggi sebelum meluncurkan instans Anda.

11. Luncurkan instance Anda:

    Saat Anda siap meluncurkan instans, di  panel Ring  kasan, pilih  Luncurkan instance.

Amazon EC2 meluncurkan instans Anda dengan cepat menggunakan pengaturan yang Anda tentukan. Jika Anda tidak menentukan pengaturan, default digunakan. Span duk Sukses mengkonfirmasi peluncuran.

Selamat! Anda telah berhasil meluncurkan instance pengujian Anda!

Tugas 5: Temukan instance Anda

Dalam tugas ini, Anda akan menemukan instance yang baru saja Anda luncurkan di konsol EC2.

Ikuti langkah-langkah berikut untuk menemukan instans Anda di konsol EC2

1. Buka halaman Instans:

   Jika Anda masih berada di halaman sukses, pilih ID instance di spanduk Sukses.

   Jika Anda telah menavigasi jauh, pilih Instans dari panel navigasi.
2. Temukan instans Anda:

   Di kolom Nama, temukan instance Anda dengan nama yang Anda berikan.

Tugas 6: Melihat konfigurasi instans

Dalam tugas ini, Anda akan terbiasa melihat detail konfigurasi instans Anda.

Ikuti langkah-langkah berikut untuk melihat konfigurasi instans Anda

1. Temukan instans Anda:

   Di kolom Nama, temukan instance Anda dengan nama yang Anda berikan.
2. Buka halaman detail instance:

   Pilih kotak centang di sebelah nama instance Anda, lalu pilih menu Tind akan (kanan atas), lalu pilih Lihat detail untuk membuka halaman detail instance tempat Anda dapat meninjau konfigurasinya.

   Dalam tutorial sebelumnya, Anda memilih tautan ID instance untuk membuka halaman detail instance. Anda akan menemukan bahwa ada lebih dari satu cara untuk menyelesaikan tugas di konsol EC2.

3. Jelajahi detail konfigurasi instance:

   Luangkan beberapa menit untuk menjelajahi detail konfigurasi instans Anda.

   Tip: Untuk menemukan bidang dengan cepat, tekan Ctrl+F atau Command+F pada keyboard Anda.

   a. AMI: Dapatkah Anda menemukan AMI yang Anda gunakan untuk meluncurkan instans Anda? Anda dapat menemukan informasi dalam AMI ID dan nama AMI pada tab Detail.
   b. Jenis instans: Dapatkah Anda menemukan jenis instans? Mungkin t3.micro, misalnya.
   c. Pasangan kunci: Dapatkah Anda menemukan pasangan kunci yang Anda pilih saat meluncurkan instans? Ini ditentukan untuk pasangan kunci yang ditetapkan saat peluncuran. Perhatikan bahwa jika Anda mengubah pasangan kunci di masa mendatang, nilai di sini tidak akan berubah.
   d. VPC: Dapatkah Anda menemukan ID VPC Anda? Anda akan menemukan semua pengaturan konfigurasi terkait jaringan di tab Jaringan. ID VPC dalam format yang mirip dengan contoh berikut: vpc-1a2b3c4d
   e. Subnet: Dapatkah Anda menemukan ID subnet tempat Anda meluncurkan instance Anda? Ini dalam format yang mirip dengan contoh berikut: subnet-1a2b3c4d
   f. Alamat IPv4 publik: Dapatkah Anda menemukan alamat IPv4 publik yang dialokasikan ke instans Anda? Ini dalam format yang mirip dengan contoh berikut: 34.24 2.148.128.
   g. Grup keamanan: Dapatkah Anda menemukan aturan masuk yang dibuat untuk mengizinkan lalu lintas SSH dari mana saja (0.0.0.0./0)? Anda akan menemukan semua pengaturan konfigurasi terkait keamanan di tab Keamanan.
   h. Penyimpanan: Dapatkah Anda menemukan volume yang dibuat untuk contoh ini? Anda akan menemukan semua pengaturan konfigurasi terkait penyimpanan di tab Storage.
   i. Tag instance: Nama yang Anda berikan pada instance Anda sebenarnya adalah tag. Dapatkah Anda menemukan tag instance Anda? Pilih tab Tanda. Kuncinya adalah Nama, dan nilainya adalah nama yang Anda berikan.
   j. Status instans: Dapatkah Anda memverifikasi status instance Anda? Seharusnya berjalan.

Luangkan beberapa menit lagi untuk menjelajahi bidang konfigurasi instance lainnya. Ketika Anda siap, lanjutkan ke tugas berikutnya.

Tugas 7: Biasakan diri Anda dengan komponen kunci untuk menghubungkan ke instance

Dalam tugas ini, Anda akan menjelajahi komponen kunci yang diperlukan untuk terhubung ke instance EC2. Ini adalah protokol koneksi, DNS publik, grup keamanan, pasangan kunci, dan nama pengguna instance.

Untuk membantu memvisualisasikan komponen-komponen ini, pikirkan menghubungkan ke contoh seperti pergi ke rumah Anda:

- Protokol koneksi — Mode transportasi Anda: Sama seperti memilih cara pulang, Anda memilih protokol koneksi yang akan membawa Anda ke instance Anda. Dalam tutorial ini, kita akan menggunakan SSH (Secure Shell), yang menciptakan terowongan aman untuk menghubungkan komputer Anda ke instance Anda melalui internet.
- DNS Publik — Alamat rumah: Sama seperti rumah Anda memiliki alamat unik, instance EC2 Anda memiliki nama DNS publik sendiri (misalnya,`ec2-18-201-118-201.eu-west-1.compute.amazonaws.com`). Nama DNS publik ini memungkinkan SSH untuk terhubung langsung ke instance Anda.
- Kelompok keamanan — Penjaga gerbang: Bay angkan rumah Anda memiliki penjaga gerbang yang mengontrol siapa yang boleh masuk atau keluar. Demikian pula, instance EC2 memiliki grup keamanan yang bertindak seperti gatekeeper, mengontrol jenis lalu lintas jaringan mana yang diizinkan masuk atau keluar dari instans Anda. Hanya lalu lintas yang Anda izinkan secara eksplisit (misalnya, lalu lintas SSH dari alamat IP komputer Anda) yang diizinkan masuk.
- Kunci pribadi — Kunci pintu depan Anda: Ketika Anda meluncurkan instance, Anda menentukan pasangan kunci. Kunci publik ditempatkan pada instance, dan Anda menyimpan kunci pribadi di komputer Anda. Kunci pribadi bertindak sebagai kunci pintu depan Anda—tanpanya, Anda tidak bisa masuk ke instance Anda.
- Nama pengguna contoh — Penduduk: Ketika Anda tiba di rumah Anda, Anda perlu mengidentifikasi diri Anda untuk membuktikan bahwa Anda adalah penduduk. Demikian pula, saat menghubungkan ke instance, Anda memberikan nama pengguna. Instans yang berbeda memiliki nama pengguna default yang berbeda, tergantung pada sistem operasinya. Misalnya, instans Amazon Linux digunakan `ec2-user` sebagai nama pengguna default.

Perintah koneksi

Untuk menyambung ke instans EC2 Anda, gunakan perintah berikut di jendela terminal:

```
ssh -i "test-instance-key-pair.pem" ec2-user@ec2-18-201-118-201.eu-
west-1.compute.amazonaws.com
```

Berikut rincian dari apa yang dilakukan perintah:

- `ssh`- Perintah ini menentukan protokol koneksi, memulai koneksi SSH (Secure Shell) ke instance Anda.
- `-i "test-instance-key-pair.pem"`- `-i` Bendera menunjukkan file kunci pribadi yang diperlukan untuk mengotentikasi koneksi. File kunci pribadi ini harus cocok dengan pasangan kunci yang Anda tentukan saat meluncurkan instance. Jika file kunci pribadi Anda disimpan dalam folder tertentu, tentukan jalur lengkap ke file tersebut.
- `ec2-user`— Ini adalah nama pengguna untuk masuk ke instance. Untuk instans Amazon Linux, nama pengguna defaultnya adalah`ec2-user`. AMI lain mungkin menggunakan nama pengguna default yang berbeda, seperti `ubuntu` untuk instance Ubuntu.
- `@`— Simbol ini memisahkan nama pengguna dari alamat instance.
- `ec2-18-201-118-201.eu-west-1.compute.amazonaws.com`— Ini adalah alamat publik instans Anda (DNS publik), yang mencakup alamat IPv4 publik dan. Wilayah AWS Ini secara unik mengidentifikasi contoh.

Apa yang terjadi ketika Anda menjalankan perintah

Setelah Anda menjalankan perintah, SSH membuat terowongan aman dan mengotentikasi dengan kunci pribadi Anda. Jika grup keamanan instans mengizinkan lalu lintas, Anda mendapatkan akses ke instans EC2 Anda. Anda sekarang dapat mengontrol instance dari komputer Anda seolah-olah Anda sedang duduk tepat di depannya. Anda dapat menjalankan perintah, menginstal perangkat lunak, dan mengelola file — seperti yang Anda lakukan di mesin lokal Anda.

Tugas 8: Hubungkan ke instance Anda

Dalam tugas ini, Anda akan terhubung ke instance Anda menggunakan klien SSH di komputer Anda. Pada tugas sebelumnya, kami memperkenalkan komponen untuk menghubungkan ke instance menggunakan analogi pergi ke rumah Anda. Sekarang, kita akan fokus untuk menghubungkan ke instance EC2 yang sebenarnya.

Ada berbagai cara untuk terhubung ke instance. Metode yang Anda gunakan untuk menghubungkan tergantung pada sistem operasi instans. Karena Anda telah meluncurkan instance Linux, Anda akan menggunakan klien SSH di komputer lokal Anda.

Pertama, periksa apakah komputer Anda memiliki klien SSH yang diinstal

Sebagian besar komputer dilengkapi dengan klien SSH yang sudah diinstal sebelumnya. Untuk memeriksa, buka jendela terminal di komputer Anda dan jalankan perintah berikut:

```
ssh
```

Jika perintah dikenali, Anda siap untuk terhubung.

Jika perintah tidak dikenali, Anda harus menginstal klien SSH. Petunjuk untuk menginstal klien SSH berada di luar cakupan tutorial ini. Jika Anda memerlukan bantuan, lihat Prasyarat koneksi SSH di panduan pengguna ini atau cari secara online untuk petunjuk tentang cara menginstal klien SSH pada sistem operasi Anda.

Ikuti langkah-langkah ini untuk menyambung ke instance Anda

1. Mulai menghubungkan:

   Jika Anda berada di halaman detail instans di konsol Amazon EC2, pilih tombol Connect (kanan atas).

   Jika Anda telah menavigasi jauh, pilih Instans dari panel navigasi. Kemudian, pada halaman Instans, pilih kotak centang di sebelah nama instance Anda dan pilih tombol Connect (kanan atas).

   Ini membuka halaman Connect to instance.
2. Pilih metode koneksi:

   Pada halaman Connect to instance, pilih tab klien SSH.

   Luangkan waktu sejenak untuk meninjau teks di halaman ini, karena ini adalah langkah-langkah yang akan Anda ikuti selanjutnya.
3. Tinjau perintah SSH:

   Di bawah Contoh, Anda akan melihat perintah yang secara otomatis dibuat dan disesuaikan dengan detail instans Anda. Nama kunci pribadi berasal dari nama kunci publik yang ditentukan saat peluncuran.

   Perintah terlihat seperti ini:

```
ssh -i "test-instance-key-pair.pem" ec2-user@ec2-18-201-118-201.eu-
west-1.compute.amazonaws.com
```

4. Salin perintah SSH:

   Pilih ikon salin di sebelah contoh perintah SSH.

5. Buka jendela terminal:

   Di komputer lokal Anda, buka jendela terminal.

6. Tempel dan jalankan perintah SSH:

   Tempelkan perintah SSH ke jendela terminal. Jika Anda menyimpan file kunci pribadi Anda di folder tertentu, edit perintah untuk menyertakan jalur file lengkap.

   Tekan Enter pada keyboard Anda.

   Anda akan melihat respons yang mirip dengan berikut ini:

```
The authenticity of host 'ec2-18-201-118-201.eu-west-1.compute.amazonaws.com
 (18-201-118-201)' can't be established.
ED25519 key fingerprint is SHA256:examplehxj9aOr1MogvKOoMNskVVIRBQBoq0example.This
 key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

7. Lengkapi koneksi:

   Masukkan **`yes`** dan tekan Return pada keyboard Anda.

   Memverifikasi sidik jari berada di luar cakupan tutorial ini. Untuk mempelajari selengkapnya, lihat (Opsional) Dapatkan sidik jari instans.

   Setelah koneksi berhasil, prompt terminal berubah untuk menampilkan DNS publik instance Anda.

Selamat!  Anda telah berhasil terhubung ke instans Anda!

Tugas 9: Hentikan instance Anda

Dalam tugas ini, Anda akan menghentikan instans untuk mempertahankan manfaat Tingkat Gratis Anda. Ketika instans Anda dihentikan, Anda berhenti mengeluarkan biaya untuk itu. Jika Anda

membuat Akun AWS sebelum 15 Juli 2025 dan memenuhi syarat untuk Tingkat Gratis, Anda akan terus dikenakan biaya untuk penyimpanan EBS.

Ikuti langkah-langkah ini untuk menghentikan instans Anda

1. Mulai berhenti:

   Jika Anda masih berada di halaman Hu bungkan ke instance, pilih Instans dari breadcrumb. Jika Anda telah menavigasi jauh, pilih Instans dari panel navigasi.

   Kemudian, pada halaman Instans, pilih kotak centang di sebelah nama instance Anda, lalu pilih menu Status Inst ance (kanan atas), lalu pilih H entikan instance. Saat diminta, pilih Stop.
2. Memantau status instans:

   Pada halaman Instans, periksa kolom Status Instance. Status instans Anda berubah menjadi Ber henti dan kemudian Dih entikan. Jika Anda tidak melihat teks lengkap, coba perlebar kolom.

   Jika menurut Anda status instance telah berubah dari Ber henti menjadi Dih entikan, tetapi Anda belum melihatnya, pilih ikon refresh (di atas tabel) untuk menyegarkan tabel Instans.

Hal utama yang dapat diambil

Dalam tutorial ini, Anda membahas konsep-konsep kunci berikut:

- AMI mengacu pada Image Mesin Amazon, yang merupakan template yang berisi sistem operasi dan perangkat lunak yang diperlukan untuk meluncurkan sebuah instance.
- Jenis instance mengacu pada perangkat keras komputer host yang digunakan untuk instance Anda. Ini menentukan CPU, memori, penyimpanan, dan kapasitas jaringan instance Anda.
- Pasangan kunci mengacu pada kumpulan kunci publik dan pribadi yang dapat Anda gunakan untuk menghubungkan instans dengan aman.
- Jaringan mengacu pada VPC (cloud pribadi virtual yang didedikasikan untuk akun Anda di dalam AWS Cloud) dan subnet (berbagai alamat IP dalam VPC Anda).
- Grup keamanan mengacu pada seperangkat aturan yang mengontrol lalu lintas apa yang dapat mencapai instans Anda.
- Volume EBS mengacu pada penyimpanan data untuk instance Anda. Setiap instance memiliki volume root untuk menyimpan AMI dan satu atau lebih volume data opsional.

- Tag adalah metadata yang dapat Anda tetapkan secara opsional ke instance Anda. Nama instance adalah tag, yang Kun cin ya adalah Nama, dan Nilai adalah pilihan Anda.
- Mengh ubungkan mengacu pada mengakses instance Anda melalui internet.
- SSH mengacu pada protokol koneksi Secure Shell yang dapat Anda gunakan untuk terhubung ke instance Anda.
- Public DNS adalah alamat publik unik instans Anda.
- Nama pengguna instans ditentukan oleh sistem operasi instans Anda dan diperlukan untuk menghubungkan.
- Mengh entikan instans Anda akan menghentikan tagihan instans, tetapi biaya penyimpanan EBS tetap berlaku.

Langkah selanjutnya

Untuk membangun kepercayaan diri dalam meluncurkan, menghubungkan, dan menghentikan instance, pertimbangkan untuk mengulangi langkah-langkah dalam tutorial ini. Pastikan untuk menghentikan instans apa pun yang Anda luncurkan untuk mempertahankan manfaat Tingkat Gratis Anda.

Setelah Anda merasa nyaman dengan dasar-dasar ini, Anda dapat menjelajahi tutorial yang lebih canggih. Untuk tutorial lainnya, lihat Mencari tutorial lain?

Jika Anda membuat video Akun AWS sebelum 15 Juli 2025, pertimbangkan untuk menonton video berdurasi 6 menit berikut: Bagaimana cara menghindari tagihan pada akun saya saat menggunakan layanan AWS Tingkat Gratis

Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, pertimbangkan untuk meninjau informasi berikut: Jelaj AWS ahi layanan dengan Tingkat AWS Gratis di Panduan AWS Billing Pengguna

## Referensi untuk parameter konfigurasi EC2 instans Amazon

Wizard instance peluncuran dan template peluncuran di EC2 konsol Amazon menyediakan semua parameter untuk mengonfigurasi EC2 instans Amazon.

Kecuali untuk key pair, wizard instance peluncuran memberikan nilai default untuk setiap parameter. Anda dapat menerima salah satu atau semua default, atau mengonfigurasi instance dengan nilai Anda sendiri. Saat membuat template peluncuran, parameternya opsional. Jika Anda menggunakan template peluncuran untuk meluncurkan instance, parameter yang ditentukan dalam template

peluncuran akan mengganti nilai default di wizard instance peluncuran. Parameter apa pun yang tidak ditentukan dalam template peluncuran akan default ke nilai yang disediakan oleh wizard instance peluncuran.

Parameter dikelompokkan dalam wizard instance peluncuran dan template peluncuran. Deskripsi berikut disajikan sesuai dengan pengelompokan parameter di konsol.

Parameter untuk konfigurasi instans

- Nama dan tanda
- Aplikasi dan Gambar OS (Gambar Mesin Amazon)
- Tipe instans
- Pasangan kunci (login)
- Pengaturan jaringan
- Mengonfigurasi penyimpanan
- Detail lanjutan
- Ringkasan

## Nama dan tanda

Nama instans adalah tanda, di mana kuncinya adalah Name, dan nilainya adalah nama yang Anda tentukan. Anda dapat menandai instance, volume, dan antarmuka jaringan. Untuk Instans Spot, Anda hanya dapat menandai permintaan Instans Spot. Untuk informasi tentang tanda, lihat Tandai sumber daya Amazon EC2 Anda.

Menentukan nama instans dan tanda tambahan bersifat opsional.

- Untuk Nama, masukkan nama deskriptif untuk instans tersebut. Jika Anda tidak menentukan nama, instans dapat diidentifikasi berdasarkan ID-nya, yang secara otomatis dihasilkan saat Anda meluncurkan instans tersebut.
- Untuk menambahkan tanda tambahan, pilih Tambahkan tanda tambahan. Pilih Tambahkan tanda, lalu masukkan kunci dan nilai, lalu pilih jenis sumber daya yang akan diberi tanda. Pilih Tambah tanda lagi untuk setiap tanda tambahan yang akan ditambahkan.

Anda hanya dapat menentukan nama instance saat meluncurkan instance. Anda tidak dapat memberi nama instance saat membuat template peluncuran, tetapi Anda dapat menambahkan tag untuk sumber daya yang dibuat saat instance diluncurkan.

## Aplikasi dan Gambar OS (Gambar Mesin Amazon)

Amazon Machine Image (AMI) berisi informasi yang diperlukan untuk membuat instans. Misalnya, AMI mungkin berisi perangkat lunak yang diperlukan untuk bertindak sebagai server web, seperti Linux, Apache, dan situs web Anda.

Anda dapat menemukan AMI yang cocok sebagai berikut. Dengan setiap opsi untuk menemukan AMI, Anda dapat memilih Batal (di kanan atas) untuk kembali ke wizard peluncuran instans tanpa memilih AMI.

Bilah pencarian

Untuk mencari semua yang tersedia AMIs, masukkan kata kunci di bilah pencarian AMI dan kemudian tekan Enter. Untuk memilih AMI, pilih Pilih.

Terbaru

AMIs Yang baru saja Anda gunakan.

Pilih Baru diluncurkan atau Saat ini sedang digunakan, kemudian pilih AMI dari Amazon Machine Image (AMI).

Saya AMIs

Pribadi AMIs yang Anda miliki, atau pribadi AMIs yang telah dibagikan dengan Anda.

Pilih Milik saya atau Dibagikan dengan saya, kemudian pilih AMI dari Amazon Machine Image (AMI).

Mulai Cepat

AMIs dikelompokkan berdasarkan sistem operasi (OS) untuk membantu Anda memulai dengan cepat.

Pertama, pilih OS yang Anda butuhkan, lalu pilih AMI dari Amazon Machine Image (AMI). Untuk memilih AMI yang dapat digunakan di bawah Tingkat AWS Gratis, pastikan AMI ditandai Tingkat Gratis yang memenuhi syarat.

Jelajahi lebih AMIs

Pilih Jelajahi lebih banyak AMIs untuk menelusuri katalog AMI lengkap.

- Untuk mencari melalui semua yang tersedia AMIs, masukkan kata kunci di bilah pencarian dan kemudian tekan Enter.

- Untuk menemukan AMI dengan menggunakan parameter Systems Manager, pilih tombol panah di sebelah kanan bilah pencarian, lalu pilih parameter Cari berdasarkan Systems Manager. Untuk informasi selengkapnya, lihat Referensi AMI menggunakan parameter Manajer Sistem.
- Untuk mencari berdasarkan kategori, pilih Mulai Cepat AMIs, Saya AMIs AWS Marketplace AMIs, atau Komunitas AMIs.

  AWS Marketplace Ini adalah toko online tempat Anda dapat membeli perangkat lunak yang berjalan AWS, termasuk AMIs. Untuk informasi selengkapnya tentang meluncurkan instance dari AWS Marketplace, lihatLuncurkan instans Amazon EC2 dari AMI AWS Marketplace.
  Di Komunitas AMIs, Anda dapat menemukan AMIs bahwa anggota AWS komunitas telah menyediakan untuk digunakan orang lain. AMIsdari Amazon atau mitra terverifikasi ditandai Penyedia terverifikasi.
- Untuk memfilter daftar AMIs, pilih satu atau beberapa kotak centang di bawah Perbaiki hasil di sebelah kiri layar. Opsi filter berbeda tergantung pada kategori pencarian yang dipilih.
- Periksa Tipe perangkat root yang terdaftar untuk setiap AMI. Perhatikan jenis AMIs yang Anda butuhkan: ebs (didukung oleh Amazon EBS) atau instance-store (didukung oleh toko instance). Untuk informasi selengkapnya, lihat Tipe volume root.
- Periksa Ttipe virtualisasi yang tercantum untuk setiap AMI. Perhatikan jenis mana AMIs yang Anda butuhkan: baik hvm atau paravirtual. Sebagai contoh, beberapa tipe instans memerlukan HVM. Untuk informasi selengkapnya tentang jenis virtualisasi Linux, lihatTipe virtualisasi.
- Periksa Mode booting yang terdaftar untuk setiap AMI. Perhatikan mana yang AMIs menggunakan mode boot yang Anda butuhkan: baik legacy-bios, uefi, atau uefi-preferen. Untuk informasi selengkapnya, lihat Perilaku peluncuran instans dengan mode boot Amazon EC2.
- Pilih AMI yang memenuhi kebutuhan Anda, lalu pilih Pilih.

Peringatan saat mengganti AMI

Ketika Anda meluncurkan instance, jika Anda mengubah konfigurasi volume atau grup keamanan apa pun yang terkait dengan AMI yang dipilih, dan kemudian Anda memilih AMI yang berbeda, sebuah jendela terbuka untuk memperingatkan Anda bahwa beberapa pengaturan Anda saat ini akan diubah atau dihapus. Anda dapat meninjau perubahan pada grup keamanan dan volume. Selanjutnya, Anda dapat memilih untuk melihat volume mana yang akan ditambahkan dan dihapus, atau hanya melihat volume yang akan ditambahkan. Peringatan ini tidak muncul saat membuat template peluncuran.

## Tipe instans

Tipe instans mendefinisikan konfigurasi perangkat keras dan ukuran instans. Tipe instans yang lebih besar memiliki lebih banyak CPU dan memori. Untuk informasi selengkapnya, lihat jenis EC2 instans Amazon.

- Tipe Instans: Pastikan bahwa tipe instans kompatibel dengan AMI yang Anda tentukan. Untuk informasi selengkapnya, lihat Tipe instans Amazon EC2.

  Tingkat Gratis — Anda dapat menggunakan jenis instans yang diberi label Tingkat gratis yang memenuhi syarat secara gratis di bawah Tingkat Gratis. Jenis instans spesifik bergantung pada saat Anda membuat Akun AWS.

  Jika Anda membuat Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan Amazon EC2 di bawah Tingkat Gratis dengan memilih jenis instans t2.micro, atau jenis instans t3.micro di Wilayah di mana t2.micro tidak tersedia. Ketahuilah bahwa ketika Anda meluncurkan instans t3.micro, default ke mode Tidak Terbatas, yang mungkin dikenakan biaya tambahan berdasarkan penggunaan CPU.

  Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan jenis instans t3.micro, t3.small, t4g.micro, t4g.small, c7i-flex.large, dan m7i-flex.large selama 6 bulan atau sampai kredit Anda habis.

  Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.
- Bandingkan jenis instans: Anda dapat membandingkan jenis instans yang berbeda dengan atribut berikut: jumlah vCPUs, arsitektur, jumlah memori (GiB), jumlah penyimpanan (GB), jenis penyimpanan, dan kinerja jaringan.
- Dapatkan saran: Anda bisa mendapatkan panduan dan saran untuk jenis instance dari pencari jenis EC2 instance. Untuk informasi selengkapnya, lihat Dapatkan rekomendasi dari pencari tipe instans EC2.
- (Hanya meluncurkan templat) Lanjutan: Untuk menentukan atribut instance dan membiarkan Amazon EC2 mengidentifikasi jenis instance dengan atribut tersebut, pilih Lanjutan, lalu pilih Tentukan atribut tipe instance.
  - Jumlah v CPUs: Masukkan jumlah minimum dan maksimum v CPUs untuk persyaratan komputasi Anda. Untuk menunjukkan tidak ada batas, masukkan minimum **0**, dan biarkan maksimum kosong.

- Jumlah memori (MiB): Masukkan jumlah memori minimum dan maksimum, dalam MiB, untuk kebutuhan komputasi Anda. Untuk menunjukkan tidak ada batas, masukkan minimum **0**, dan biarkan maksimum kosong.
- Perluas atribut tipe instans opsional dan pilih Tambahkan atribut untuk mengekspresikan persyaratan komputasi Anda secara lebih detail. Untuk informasi tentang setiap atribut, lihat InstanceRequirementsRequestdi Referensi Amazon EC2 API.
- Tipe instans yang dihasilkan: Anda dapat melihat pratinjau tipe instans yang cocok dengan atribut yang ditentukan. Untuk mengecualikan tipe instans, pilih Tambahkan atribut, dan dari daftar Atribut, pilih Tipe instans yang dikecualikan. Dari daftar Nilai atribut, pilih tipe instans yang akan dikecualikan.

## Pasangan kunci (login)

Untuk Nama pasangan kunci, pilih pasangan kunci yang ada, atau pilih Buat pasangan kunci baru untuk membuat yang baru. Untuk informasi selengkapnya, lihat Pasangan kunci Amazon EC2 dan instans Amazon EC2.

> **Important**
>
> Jika Anda memilih opsi Lanjutkan tanpa pasangan kunci (Tidak direkomendasikan), Anda tidak akan dapat terhubung ke instans tersebut, kecuali Anda memilih sebuah AMI yang dikonfigurasi agar pengguna dapat masuk dengan cara lain.

## Pengaturan jaringan

Pengaturan jaringan menentukan alamat IP, grup keamanan, dan antarmuka jaringan untuk instance Anda. Anda dapat menggunakan pengaturan jaringan default atau mengkonfigurasinya sesuai kebutuhan.

- (Hanya Luncurkan wizard instance) VPC: Pilih VPC yang ada untuk instance Anda. VPC default untuk Wilayah dipilih secara default. Atau, Anda dapat memilih VPC yang Anda buat atau yang dibagikan dengan Anda. Untuk informasi selengkapnya, lihat Virtual private cloud untuk instans EC2.
- Subnet: Pilih subnet untuk instans Anda atau pilih Buat subnet baru untuk membuat subnet baru menggunakan konsol Amazon VPC.

  - Anda dapat membuat subnet di Availability Zone, Local Zone, Wavelength Zone, atau Outpost Zone untuk VPC yang dipilih.
  - Untuk meluncurkan instance di subnet IPv6 -only, instance harus berupa instance berbasis Nitro.
- (Hanya luncurkan wizard instans) Tetapkan otomatis IP Publik: Aktifkan atau nonaktifkan penetapan otomatis alamat publik. IPv4 Saat meluncurkan instance ke subnet default, nilai defaultnya adalah Aktifkan. Saat meluncurkan instance ke subnet nondefault, nilai defaultnya adalah Nonaktifkan. Untuk informasi selengkapnya, lihat Alamat IPv4 publik.

  Anda tidak dapat mengaktifkan opsi ini untuk subnet nondefault jika Anda menambahkan antarmuka jaringan sekunder. Untuk informasi selengkapnya, lihat the section called “Tetapkan alamat IPv4 publik saat peluncuran”.
- (Hanya luncurkan wizard instance) Tetapkan otomatis IPv6 IP: Aktifkan atau nonaktifkan penetapan alamat secara otomatis. IPv6 Untuk informasi selengkapnya, lihat Alamat IPv6.
- Firewall (grup keamanan): Pilih grup keamanan yang ada atau buat yang baru. Pastikan grup keamanan Anda memiliki aturan yang memungkinkan lalu lintas ke dan dari instans Anda. Semua lalu lintas lainnya diabaikan.

  Jika Anda membuat grup keamanan baru, kami secara otomatis membuat aturan masuk yang memungkinkan Anda terhubung ke instans Anda dari semua alamat IP melalui SSH (instance Linux) atau RDP (instance Windows. Anda dapat menghapus atau memodifikasi aturan ini sesuai kebutuhan. Anda dapat menambahkan aturan sesuai kebutuhan. Untuk informasi selengkapnya, lihat Mengonfigurasi aturan grup keamanan.

  > ⚠ Warning
  >
  > Aturan yang memungkinkan semua alamat IP untuk mengakses instans Anda melalui SSH atau RDP dapat diterima jika Anda meluncurkan instance pengujian sebentar dan akan menghentikan atau menghentikannya setelah waktu yang singkat. Mereka tidak aman untuk lingkungan produksi. Anda harus mengotorisasi hanya rentang alamat IP tertentu untuk mengakses instans Anda.

  Grup keamanan ini ditambahkan ke antarmuka jaringan utama dan antarmuka jaringan sekunder apa pun. Anda dapat memilih grup keamanan tambahan untuk antarmuka jaringan Anda, tetapi Anda tidak dapat menghapus salah satu yang Anda pilih di sini.

- Konfigurasi jaringan lanjutan - Anda dapat mengkonfigurasi antarmuka jaringan utama sesuai kebutuhan. Untuk menambahkan antarmuka jaringan sekunder, pilih Tambahkan antarmuka jaringan. Jumlah antarmuka jaringan yang dapat Anda tambahkan tergantung pada jenis instance yang Anda pilih. Perhatikan bahwa bagian ini hanya tersedia jika Anda memilih subnet.
  - Indeks perangkat: Indeks perangkat. Antarmuka jaringan utama harus ditetapkan ke indeks 0.
  - Antarmuka jaringan: Antarmuka jaringan. Pilih Antarmuka baru untuk memungkinkan Amazon EC2 membuat antarmuka baru, atau pilih antarmuka jaringan yang ada dan tersedia. Jika Anda memilih antarmuka jaringan yang ada sebagai antarmuka jaringan utama, Anda tidak dapat mengaktifkan Auto-assign IP Publik untuk subnet nondefault.
  - Deskripsi: Deskripsi untuk antarmuka jaringan baru.
  - Subnet: Subnet tempat membuat antarmuka jaringan baru. Instance diluncurkan di subnet yang sama dengan antarmuka jaringan utama.

    Anda harus memilih subnet untuk antarmuka jaringan sekunder dari Availability Zone yang sama dengan subnet untuk antarmuka jaringan utama. Jika Anda memilih subnet dari VPC lain, label Multi-VPC muncul di sebelah antarmuka jaringan. Ini memungkinkan Anda membuat instance multi-homed VPCs dengan konfigurasi jaringan dan keamanan yang berbeda.

    Untuk meluncurkan EC2 instance ke subnet IPv6 -only, Anda harus menggunakan instance berbasis Nitro. Saat meluncurkan instance IPv6 -only, ada kemungkinan bahwa DHCPv6 mungkin tidak segera menyediakan instance dengan server nama IPv6 DNS. Selama penundaan awal ini, instance mungkin tidak menyelesaikan domain publik. Anda dapat mengubah file konfigurasi dan gambar ulang AMI Anda sehingga file tersebut memiliki alamat server nama IPv6 DNS segera saat booting.
  - Grup keamanan: Grup keamanan untuk diasosiasikan dengan antarmuka jaringan. Anda harus memilih grup keamanan dari VPC yang sama dengan subnet untuk antarmuka jaringan.
  - (Hanya meluncurkan templat) Tetapkan IP publik secara otomatis: Tentukan apakah instans Anda menerima alamat publik IPv4 . Secara default, instance di subnet default menerima IPv4 alamat publik dan instance di subnet nondefault tidak. Anda dapat memilih Aktifkan atau Nonaktifkan untuk mengganti pengaturan default subnet. Untuk informasi selengkapnya, lihat Alamat IPv4 publik.
  - IP Primer: IPv4 Alamat pribadi dari kisaran subnet Anda. Biarkan kosong untuk membiarkan Amazon EC2 memilih IPv4 alamat pribadi untuk Anda.
  - IP Sekunder: IPv4 Alamat pribadi tambahan dari kisaran subnet Anda. Pilih Tetapkan secara manual dan masukkan IPv4 alamat. Pilih Tambahkan IP untuk menambahkan IPv4 alamat lain.

Atau, pilih Tetapkan secara otomatis dan masukkan nilai untuk menunjukkan jumlah IPv4 alamat yang EC2 dipilih Amazon untuk Anda.

- (IPv6-only) IPv6 IPs: IPv6 alamat dari kisaran subnet. Pilih Tetapkan secara manual dan masukkan IPv6 alamat. Pilih Tambahkan IP untuk menambahkan IPv6 alamat lain. Atau, pilih Tetapkan secara otomatis dan masukkan nilai untuk menunjukkan jumlah IPv6 alamat yang EC2 dipilih Amazon untuk Anda.
- IPv4 Awalan: IPv4 Awalan untuk antarmuka jaringan. Pilih Tetapkan secara manual dan masukkan IPv4 awalan. Atau, pilih Tetapkan secara otomatis dan masukkan nilai untuk menunjukkan jumlah IPv4 awalan yang EC2 dipilih Amazon untuk Anda.
- IPv6 Awalan: IPv6 Awalan untuk antarmuka jaringan. Pilih Tetapkan secara manual dan masukkan IPv6 awalan. Atau, pilih Tetapkan secara otomatis dan masukkan nilai untuk menunjukkan jumlah IPv6 awalan yang EC2 dipilih Amazon untuk Anda.
- (Dual-stack dan IPv6 -only) Tetapkan IPv6 IP Primer: Jika Anda memilih subnet dual-stack atau IPv6 -only, tetapkan alamat utama. IPv6 Ini membantu mencegah gangguan lalu lintas ke instance atau antarmuka jaringan. Aktifkan opsi ini jika Anda mengandalkan IPv6 alamat yang tidak berubah. Anda tidak dapat menghapus IPv6 alamat utama nanti. Ketika Anda mengaktifkan alamat IPv6 GUA menjadi primer IPv6, IPv6 GUA pertama menjadi IPv6 alamat utama sampai instance dihentikan atau antarmuka jaringan terlepas. Jika Anda memiliki beberapa IPv6 alamat yang terkait dengan antarmuka jaringan dan Anda membiarkan Amazon EC2 menetapkan IPv6 alamat utama, alamat IPv6 GUA pertama yang terkait dengan antarmuka jaringan adalah IPv6 alamat utama.
- Hapus saat penghentian: Menunjukkan apakah antarmuka jaringan dihapus saat instance dihapus.
- Jenis antarmuka: Jenis antarmuka jaringan:
  - ENA: Antarmuka jaringan berkinerja tinggi yang dirancang untuk menangani throughput dan packet-per-second tingkat tinggi untuk TCP/IP protokol sambil meminimalkan penggunaan CPU. Ini adalah nilai default. Untuk informasi lebih lanjut tentang ENA, lihat Adaptor Jaringan Elastis.
  - EFA dengan ENA: Antarmuka jaringan yang mendukung perangkat ENA dan EFA untuk transportasi TCP/IP berbasis tradisional bersama dengan transportasi berbasis SRD. Untuk informasi lebih lanjut tentang EFA, lihat Adaptor Kain Elastis.
  - Khusus EFA: Antarmuka jaringan berkinerja tinggi yang dirancang untuk menangani throughput tinggi, komunikasi antar-node latensi rendah untuk transportasi berbasis SRD sambil melewati tumpukan sistem operasi. Antarmuka jaringan khusus EFA tidak mendukung alamat IP. Untuk informasi lebih lanjut tentang EFA, lihat Adaptor Kain Elastis.

- Elastic Fabric Adapter: Menunjukkan apakah antarmuka jaringan adalah Elastic Fabric Adapter. Untuk informasi selengkapnya, lihat Adaptor Kain Elastis untuk AI/ML dan beban kerja HPC di Amazon EC2.
- Indeks Kartu Jaringan: Indeks kartu jaringan. Antarmuka jaringan primer harus ditetapkan ke indeks kartu jaringan 0. Beberapa tipe instans mendukung banyak kartu jaringan.
- ENA Express: ENA Express didukung oleh teknologi AWS Scalable Reliable Datagram (SRD). Teknologi SRD menggunakan mekanisme penyemprotan paket untuk mendistribusikan beban dan menghindari kemacetan jaringan. Mengaktifkan ENA Ekspres memungkinkan instans yang didukung untuk berkomunikasi menggunakan SRD di atas lalu lintas TCP reguler bila memungkinkan. Wizard instance peluncuran atau template peluncuran tidak menyertakan konfigurasi ENA Express untuk instance kecuali Anda memilih Aktifkan atau Nonaktifkan dari daftar.
- ENA Express UDP: Jika Anda telah mengaktifkan ENA Ekspres, Anda dapat menggunakannya secara opsional untuk lalu lintas UDP. Wizard instance peluncuran atau template peluncuran tidak menyertakan konfigurasi ENA Express untuk instance kecuali Anda memilih Aktifkan atau Nonaktifkan.

## Mengonfigurasi penyimpanan

AMI yang Anda pilih mencakup satu atau lebih volume penyimpanan, termasuk volume root. Anda dapat menentukan volume tambahan untuk dilampirkan ke instans.

(Hanya luncurkan wizard contoh) Anda dapat menggunakan tampilan Sederhana atau Lanjutan. Dengan tampilan Sederhana, Anda menentukan ukuran dan tipe volume. Untuk menentukan semua parameter volume, pilih tampilan Lanjutan (di kanan atas kartu).

Dengan menggunakan tampilan Lanjutan, Anda dapat mengonfigurasi setiap volume sebagai berikut:

- Tipe penyimpanan: Pilih volume Amazon EBS atau penyimpanan instans untuk dikaitkan dengan instans Anda. Tipe volume yang tersedia di daftar tergantung pada tipe instans yang telah Anda pilih. Untuk informasi selengkapnya, lihat Penyimpanan blok sementara instans untuk instans EC2 dan volume Amazon EBS.
- Nama perangkat: Pilih dari daftar nama perangkat yang tersedia untuk volume.
- Snapshot: Pilih snapshot yang akan digunakan untuk memulihkan volume. Anda dapat mencari snapshot bersama dan publik yang tersedia dengan memasukkan teks ke dalam bidang Snapshot.
- Ukuran (GiB): Untuk volume EBS, Anda dapat menentukan ukuran penyimpanan.

- Tipe volume: Untuk volume EBS, pilih tipe volume. Untuk informasi selengkapnya, lihat Jenis volume Amazon EBS di Panduan Pengguna Amazon EBS.
- IOPS: Jika Anda telah memilih jenis volume io1, io2, atau gp3, maka Anda dapat memasukkan jumlah I/O operasi per detik (IOPS) yang dapat didukung volume. Diperlukan untuk volume io1, io2, dan gp3. Tidak didukung untuk gp2, st1, sc1, atau volume standar. Jika Anda menghilangkan paramater ini untuk templat peluncuran, Anda harus menentukan nilai untuk itu saat Anda meluncurkan sebuah instans dari templat peluncuran.
- Hapus saat pengakhiran: Untuk volume Amazon EBS, pilih Ya untuk menghapus volume saat instans diakhiri, atau pilih Tidak untuk mempertahankan volume. Untuk informasi selengkapnya, lihat Pertahankan data saat instans diakhiri.
- Terenkripsi: Jika tipe instans mendukung enkripsi EBS, Anda dapat memilih Ya untuk mengaktifkan enkripsi untuk volume tersebut. Jika Anda telah mengaktifkan enkripsi secara default di Wilayah ini, enkripsi diaktifkan untuk Anda. Untuk informasi selengkapnya, lihat enkripsi Amazon EBS di Panduan Pengguna Amazon EBS.
- Kunci KMS: Jika Anda memilih Ya untuk Terenkripsi, maka Anda harus memilih kunci yang dikelola pelanggan untuk digunakan untuk mengenkripsi volume. Jika Anda telah mengaktifkan enkripsi secara default di Wilayah ini, kunci yang dikelola pelanggan secara default akan dipilihkan untuk Anda. Anda dapat memilih kunci yang berbeda atau menentukan ARN dari kunci yang dikelola pelanggan mana pun yang Anda buat.
- Throughput: Jika Anda telah memilih jenis gp3 volume, maka Anda dapat memasukkan throughput, dalam MIB/s, yang dapat didukung volume.
- Tingkat inisialisasi volume: Jika Anda telah memilih snapshot, Anda dapat menentukan tingkat inisialisasi volume secara opsional, di Mib/s, di mana blok snapshot akan diunduh dari Amazon S3 ke volume. Untuk informasi selengkapnya, lihat Menggunakan Tarif yang Disediakan Amazon EBS untuk Inisialisasi Volume. Untuk menggunakan tingkat inisialisasi default atau pemulihan snapshot cepat (jika diaktifkan untuk snapshot yang dipilih), jangan tentukan tarif.
- Sistem file: Pasang sistem FSx file Amazon EFS atau Amazon ke instans. Untuk informasi selengkapnya tentang pemasangan sistem file Amazon EFS, lihat Menggunakan Amazon EFS dengan instans Amazon EC2 Linux. Untuk informasi selengkapnya tentang pemasangan sistem FSx file Amazon, lihat Gunakan Amazon FSx dengan instans Amazon EC2

## Detail lanjutan

Untuk Detail lanjutan, perluas bagian untuk melihat kolom dan menentukan parameter tambahan apa pun untuk instans.

- (Hanya panduan peluncuran instance) Direktori gabungan domain: Pilih Directory Service direktori (domain) tempat instance Anda bergabung setelah peluncuran. Jika Anda memilih domain, Anda harus memilih peran IAM dengan izin yang diperlukan. Untuk informasi selengkapnya tentang penggabungan domain, lihat Menggabungkan instans Amazon EC2 Linux dengan mulus ke direktori AD Microsoft AWS Terkelola (instance Linux) dan bergabung dengan instans Amazon EC2 Windows ke direktori AD AWS Microsoft Terkelola (instance Windows) dengan mulus.
- Profil instans IAM: Pilih profil instans IAM untuk dikaitkan dengan instance. Ini adalah wadah untuk peran IAM. Untuk informasi selengkapnya, lihat IAM role untuk Amazon EC2.
- Jenis nama host: Pilih apakah nama host OS tamu dari instans akan menyertakan nama sumber daya atau nama IP. Untuk informasi selengkapnya, lihat Nama host dan domain instans EC2.
- Nama Host DNS: Menentukan apakah permintaan DNS ke nama sumber daya atau nama IP (tergantung pada apa yang Anda pilih untuk jenis Hostname) akan merespons dengan alamat (catatan A), IPv4 alamat (catatan AAAA), IPv6 atau keduanya. Untuk informasi selengkapnya, lihat Nama host dan domain instans EC2.
- Pemulihan otomatis instans: Saat diaktifkan, pulihkan instans Anda jika pemeriksaan status sistem gagal. Pengaturan ini diaktifkan secara default saat peluncuran untuk jenis instans yang didukung. Untuk informasi selengkapnya, lihat Mengkonfigurasi pemulihan otomatis yang disederhanakan pada instans Amazon EC2.
- Perilaku pematian: Pilih apakah instans harus berhenti atau diakhiri saat dimatikan. Untuk informasi selengkapnya, lihat Ubah perilaku shutdown yang diprakarsai instans.
- Berhenti - Perilaku hibernasi: Untuk mengaktifkan hibernasi, pilih Aktifkan. Bidang ini hanya tersedia jika instans Anda memenuhi prasyarat hibernasi. Untuk informasi selengkapnya, lihat Hibernasi instans Amazon EC2 Anda.
- Perlindungan pengakhiran: Untuk mencegah pengakhiran yang tidak disengaja, pilih Aktifkan. Untuk informasi selengkapnya, lihat Ubah perlindungan penghentian instans.
- Perlindungan penghentian: Untuk mencegah penghentian yang tidak disengaja, pilih Aktifkan. Untuk informasi selengkapnya, lihat Aktifkan perlindungan penghentian.
- CloudWatch Pemantauan terperinci: Pilih Aktifkan untuk mengaktifkan pemantauan mendetail instans Anda menggunakan Amazon CloudWatch. Biaya tambahan berlaku. Untuk informasi selengkapnya, lihat Memantau instans Anda menggunakan CloudWatch.
- Spesifikasi kredit: Pilih Tak Terbatas agar aplikasi dapat melonjak di atas acuan selama diperlukan. Bidang ini hanya valid untuk instans T. Biaya tambahan mungkin berlaku. Untuk informasi selengkapnya, lihat Instance performa yang dapat melonjak.

- Grup penempatan: Tentukan grup penempatan untuk meluncurkan instance. Anda dapat memilih grup penempatan yang sudah ada, atau membuat grup yang baru. Tidak semua tipe instans mendukung peluncuran instans dalam grup penempatan. Untuk informasi selengkapnya, lihat Grup penempatan untuk instans Amazon EC2 Anda.
- Instans dengan pengoptimalan EBS: Instans yang dioptimalkan untuk Amazon EBS menggunakan tumpukan konfigurasi yang dioptimalkan dan menyediakan kapasitas khusus tambahan untuk I/O Amazon EBS. Jika tipe instans mendukung fitur ini, pilih Aktifkan untuk mengaktifkannya. Biaya tambahan berlaku. Untuk informasi selengkapnya, lihat the section called “Optimisasi EBS”.
- Konfigurasi bandwidth instans: Anda dapat meningkatkan bandwidth jaringan atau bandwidth EBS Anda. Hanya untuk jenis instance yang didukung. Untuk informasi selengkapnya, lihat Konfigurasi pembobotan bandwidth instans EC2.
- Opsi pembelian: Pilih Instans Spot untuk meminta Instans Spot pada harga Spot, dibatasi pada harga Sesuai Permintaan, dan pilih opsi Sesuaikan Instans Spot untuk mengubah pengaturan Instans Spot default. Anda dapat menetapkan harga maksimum (tidak disarankan), dan mengubah tipe permintaan, durasi permintaan, dan perilaku interupsi. Jika Anda tidak meminta Instans Spot, Amazon akan EC2 meluncurkan Instans Sesuai Permintaan secara default. Untuk informasi selengkapnya, lihat Mengelola Instans Spot Anda.
- Reservasi Kapasitas: Tentukan apakah akan meluncurkan instans ke Reservasi Kapasitas apa pun yang terbuka (Open), Reservasi Kapasitas tertentu (Target berdasarkan ID), atau grup Reservasi Kapasitas (Target berdasarkan group). Untuk menentukan bahwa Reservasi Kapasitas tidak boleh digunakan, pilih Tidak Ada. Untuk informasi selengkapnya, lihat Luncurkan instans ke dalam Reservasi Kapasitas yang ada.
- Penghunian: Pilih apakah akan menjalankan instans Anda pada perangkat keras bersama (Dibagikan), perangkat keras terisolasi dan khusus (Khusus), atau pada Host Khusus (Host Khusus). Jika Anda memilih untuk meluncurkan instans ke Host Khusus, Anda dapat menentukan apakah akan meluncurkan instans ke grup sumber daya host atau Anda dapat menargetkan Host Khusus tertentu. Biaya tambahan mungkin berlaku. Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2 dan Host Khusus Amazon EC.
- ID disk RAM: (Hanya berlaku untuk paravirtual (PV) AMIs) Pilih disk RAM untuk instance. Jika Anda telah memilih kernel, Anda mungkin perlu memilih RAM disk tertentu dengan driver untuk mendukungnya.
- ID Kernel: (Hanya berlaku untuk paravirtual (PV) AMIs) Pilih kernel untuk instance.
- Nitro Enclave: Memungkinkan Anda membuat lingkungan eksekusi terisolasi, yang disebut enclaves, dari instans Amazon. EC2 Pilih Aktifkan untuk mengaktifkan instance untuk AWS Nitro

Enclave. Untuk informasi lebih lanjut, lihat Apa itu Enklaf AWS Nitro? di Panduan Pengguna AWS Nitro Enclave.

- Konfigurasi lisensi: Anda dapat meluncurkan instans berdasarkan konfigurasi lisensi yang ditentukan untuk melacak penggunaan lisensi Anda. Untuk informasi selengkapnya, lihat Buat konfigurasi lisensi dalam Panduan Pengguna AWS License Manager.
- Tentukan opsi CPU: Di wizard instance peluncuran, bidang ini hanya terlihat jika jenis instans yang dipilih mendukung penentuan opsi CPU. Pilih Tentukan opsi CPU untuk menentukan nomor kustom v CPUs selama peluncuran. Atur jumlah inti CPU dan thread per inti. Untuk informasi selengkapnya, lihat Opsi CPU untuk instans Amazon EC2.
- Metadata dapat diakses: Anda dapat mengaktifkan atau menonaktifkan akses ke Layanan Metadata Instans (IMDS). Untuk informasi selengkapnya, lihat Mengonfigurasi opsi metadata instans untuk instans baru.
- IPv6 Titik akhir metadata: Anda dapat mengaktifkan instance untuk menggunakan IPv6 alamat `[fd00:ec2::254]` IMDS untuk mengambil metadata instance. Opsi ini hanya tersedia jika Anda meluncurkan instance berbasis Nitro ke subnet yang IPv6 didukung -support (tumpukan ganda atau hanya). IPv6 Untuk informasi selengkapnya tentang pengambilan metadata instans, lihat Mengakses metadata instans untuk instans EC2.
- Versi metadata: Jika Anda mengaktifkan akses ke IMDS, Anda dapat memilih untuk meminta penggunaan Layanan Metadata Instans Versi 2 saat meminta metadata instans. Untuk informasi selengkapnya, lihat Mengonfigurasi opsi metadata instans untuk instans baru.
- Batas lompatan respons metadata: Jika Anda mengaktifkan IMDS, maka Anda dapat mengatur jumlah lompatan jaringan yang diizinkan untuk token metadata. Untuk informasi selengkapnya, lihat Mengonfigurasi opsi metadata instans untuk instans baru.
- Izinkan tanda dalam metadata: Jika Anda memilih Aktifkan, instans akan mengizinkan akses ke semua tanda dari metadatanya. Jika tidak ada nilai yang ditentukan, maka secara default, akses ke tanda dalam metadata instans tidak diperbolehkan. Untuk informasi selengkapnya, lihat Aktifkan akses ke tag dalam metadata instance.
- Data pengguna: Anda dapat menentukan data pengguna untuk mengonfigurasi instans selama peluncuran, atau untuk menjalankan skrip konfigurasi. Untuk informasi selengkapnya tentang data pengguna untuk instance Linux, lihatJalankan perintah saat Anda meluncurkan instance EC2 dengan input data pengguna. Untuk informasi selengkapnya tentang data pengguna untuk instance Windows, lihatCara Amazon EC2 menangani data pengguna untuk instans Windows.

## Ringkasan

Gunakan panel Ringkasan untuk menentukan jumlah instans yang akan diluncurkan, untuk meninjau konfigurasi instans Anda, dan untuk meluncurkan instans.

- Jumlah instans: Masukkan jumlah instans yang akan diluncurkan. Semua instans akan diluncurkan dengan konfigurasi yang sama.

> **Tip**
>
> Untuk memastikan instans diluncurkan lebih cepat, bagi permintaan besar menjadi beberapa kelompok yang lebih kecil. Misalnya, buat lima permintaan peluncuran terpisah untuk masing-masing 100 instans, bukan satu permintaan peluncuran untuk 500 instans.

- (Opsional) Jika Anda menentukan lebih dari satu instance, untuk membantu memastikan bahwa Anda mempertahankan jumlah instans yang benar untuk menangani permintaan pada aplikasi Anda, Anda dapat memilih mempertimbangkan EC2 Auto Scaling untuk membuat template peluncuran dan grup Auto Scaling. Auto Scaling menskalakan jumlah instans dalam grup sesuai dengan spesifikasi Anda. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon EC2 Auto Scaling.

> **Note**
>
> Jika Amazon EC2 Auto Scaling menandai instance yang ada di grup Auto Scaling sebagai tidak sehat, instans secara otomatis dijadwalkan untuk diganti di mana instans dihentikan dan instans lainnya diluncurkan, dan Anda kehilangan data pada instans asli. Sebuah instans ditandai sebagai tidak sehat jika Anda menghentikan atau melakukan boot ulang instans, atau jika peristiwa lain menandai instans sebagai tidak sehat. Untuk informasi selengkapnya, lihat Health memeriksa instans di grup Auto Scaling di Panduan Pengguna Amazon Auto EC2 Scaling.

- Tinjau detail instans Anda, dan buat perubahan yang diperlukan. Anda dapat menavigasi langsung ke bagian dengan memilih tautannya di panel Ringkasan.
- Ketika Anda siap untuk meluncurkan instans Anda, pilih Launch instans.

# Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol

Anda dapat meluncurkan instans Amazon EC2 menggunakan wizard instans peluncuran di konsol Amazon EC2. Wizard menyediakan nilai default untuk parameter peluncuran, yang dapat Anda terima atau modifikasi agar sesuai dengan kebutuhan Anda. Satu-satunya parameter yang tidak ditentukan adalah key pair. Jika Anda memilih untuk menerima nilai default, Anda dapat dengan cepat meluncurkan instance dengan memilih hanya key pair.

> **Important**
>
> Anda dikenakan biaya untuk instance saat instance dalam `running` keadaan, bahkan jika itu tetap menganggur. Namun, jika Anda memenuhi syarat untuk Tingkat Gratis, Anda mungkin tidak dikenakan biaya. Untuk informasi selengkapnya, lihat Lacak penggunaan Tingkat Gratis Anda untuk Amazon EC2.

Untuk deskripsi setiap parameter dalam wizard instance peluncuran, lihatReferensi untuk parameter konfigurasi EC2 instans Amazon.

Topik

- Meluncurkan instans dengan cepat
- Luncurkan sebuah instans menggunakan parameter yang ditentukan

## Meluncurkan instans dengan cepat

Untuk menyiapkan instans dengan cepat untuk tujuan pengujian, ikuti langkah-langkah ini. Anda akan memilih sistem operasi dan pasangan kunci Anda, serta menerima nilai default. Kecuali untuk pasangan kunci, wizard peluncuran instans memberikan nilai default untuk semua parameter. Anda dapat menerima salah satu atau semua default, atau mengonfigurasi instans dengan menentukan nilai Anda sendiri untuk setiap parameter.

Untuk deskripsi setiap parameter dalam wizard instance peluncuran, lihatReferensi untuk parameter konfigurasi EC2 instans Amazon.

Untuk meluncurkan instance dengan cepat menggunakan wizard instance peluncuran

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/

2. Di bilah navigasi di bagian atas layar, AWS Wilayah saat ini ditampilkan (misalnya, AS Timur (Ohio)). Jika diperlukan, pilih Wilayah yang berbeda untuk meluncurkan instance.
3. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.
4. (Opsional) Pada Nama dan tanda, untuk Nama, masukkan nama deskriptif untuk instans Anda.
5. Di bawah Gambar Aplikasi dan OS (Amazon Machine Image), pilih Mulai Cepat, lalu pilih sistem operasi (OS) untuk instans Anda.
6. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang ada atau buat yang baru.
7. Di panel Ringkasan, pilih Luncurkan instans.

## Luncurkan sebuah instans menggunakan parameter yang ditentukan

Jika Anda meluncurkan instance yang akan Anda gunakan dalam produksi, Anda harus mengonfigurasi instance agar sesuai dengan kebutuhan Anda. Untuk deskripsi setiap parameter dalam wizard instance peluncuran, lihatReferensi untuk parameter konfigurasi EC2 instans Amazon.

Untuk meluncurkan instance dengan mendefinisikan semua parameter peluncuran menggunakan wizard instance peluncuran

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/
2. Di bilah navigasi di bagian atas layar, AWS Wilayah saat ini ditampilkan (misalnya, AS Timur (Ohio)). Jika diperlukan, pilih Wilayah yang berbeda untuk meluncurkan instance.
3. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.
4. (Opsional) Di bawah Nama dan tag, untuk Nama, masukkan nama deskriptif untuk instance Anda sehingga Anda dapat dengan mudah melacaknya.

   Nama instans adalah tanda, di mana kuncinya adalah Name, dan nilainya adalah nama yang Anda tentukan.
5. Di bawah Application and OS Images (Amazon Machine Image), pilih sistem operasi (OS) untuk instans Anda, lalu pilih AMI.

   AMI adalah template yang berisi sistem operasi dan perangkat lunak yang diperlukan untuk meluncurkan instance Anda.
6. Di bawah jenis Instance, pilih jenis instance.

Jenis instans menentukan konfigurasi perangkat keras (CPU, memori, penyimpanan, dan kapasitas jaringan) dan ukuran komputer host yang digunakan untuk instance Anda.

Jika Anda tidak yakin jenis instance mana yang harus dipilih, Anda dapat melakukan hal berikut:

- Pilih Bandingkan jenis instans untuk membandingkan jenis instans yang berbeda dengan atribut berikut: jumlah vCPUs, arsitektur, jumlah memori (GiB), jumlah penyimpanan (GB), jenis penyimpanan, dan kinerja jaringan.
- Pilih Dapatkan saran untuk mendapatkan panduan dan saran untuk jenis instans dari pencari jenis instans EC2. Untuk informasi selengkapnya, lihat Dapatkan rekomendasi dari pencari tipe instans EC2.

> **Note**
>
> Bergantung pada saat Anda membuat akun, Anda mungkin dapat menggunakan jenis instans secara gratis di bawah Tingkat Gratis. Jenis instans ini diberi label Tingkat gratis yang memenuhi syarat.
> Jika Anda membuat Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan Amazon EC2 di bawah Tingkat Gratis dengan memilih jenis instans t2.micro, atau jenis instans t3.micro di Wilayah di mana t2.micro tidak tersedia. Ketahuilah bahwa ketika Anda meluncurkan instans t3.micro, default ke mode Tidak Terbatas, yang mungkin dikenakan biaya tambahan berdasarkan penggunaan CPU.
> Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan jenis instans t3.micro, t3.small, t4g.micro, t4g.small, c7i-flex.large, dan m7i-flex.large selama 6 bulan atau sampai kredit Anda habis.
> Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.

7. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang sudah ada atau buat yang baru. Jika Anda tidak memerlukan key pair untuk terhubung ke instans Anda, Anda dapat memilih Proceed without a key pair (tidak disarankan).
8. Di bawah Pengaturan jaringan, Anda dapat menyimpan default jika meluncurkan instance pengujian. Jika Anda meluncurkan instans produksi, praktik terbaik adalah mengontrol lalu lintas masuk dan keluar dari instans Anda menggunakan pengaturan jaringan dan grup keamanan yang Anda tentukan.

9. Di bawah Konfigurasi penyimpanan, Anda dapat menyimpan default atau menentukan penyimpanan tambahan. AMI yang Anda pilih mencakup satu atau lebih volume penyimpanan, termasuk volume root. Anda dapat menentukan volume tambahan untuk dilampirkan ke instans.

   Anda dapat menggunakan tampilan Sederhana atau Lanjutan. Dengan tampilan Sederhana, Anda menentukan ukuran dan tipe volume. Untuk menentukan semua parameter volume, pilih tampilan Lanjutan (di kanan atas kartu).
10. Untuk detail lanjutan, perluas bagian untuk melihat bidang dan tentukan parameter tambahan apa pun untuk instance Anda.
11. Di panel Ringkasan, Anda dapat melakukan hal berikut:

    a. Tentukan jumlah instance yang akan diluncurkan.
    b. Tinjau konfigurasi instans Anda, dan arahkan langsung ke bagian dengan memilih tautannya.
    c. Ketika Anda siap untuk meluncurkan instans Anda, pilih Launch instans.

    Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.
12. (Opsional) Anda dapat membuat peringatan penagihan untuk instans tersebut. Pada layar konfirmasi, pada Langkah Berikutnya, pilih Buat peringatan tagihan dan ikuti petunjuknya. Peringatan penagihan juga dapat dibuat setelah Anda meluncurkan instans. Untuk informasi selengkapnya, lihat Membuat alarm penagihan untuk memantau perkiraan AWS tagihan Anda di Panduan CloudWatch Pengguna Amazon.

## Luncurkan instans EC2 menggunakan template peluncuran

Template peluncuran Amazon EC2 menyimpan parameter peluncuran instance sehingga Anda tidak perlu menentukannya setiap kali meluncurkan instance.

Beberapa layanan peluncuran instans secara opsional dapat menggunakan templat peluncuran saat meluncurkan instans, sementara untuk layanan lain, seperti EC2 Fleet, instans tidak dapat diluncurkan kecuali templat peluncuran digunakan. Topik ini menjelaskan cara menggunakan templat peluncuran saat meluncurkan instans menggunakan wizard peluncuran instans EC2, Amazon EC2 Auto Scaling, Armada EC2, dan Armada Spot.

Untuk informasi selengkapnya tentang template peluncuran, termasuk cara membuat template peluncuran, lihatSimpan parameter peluncuran instans di templat peluncuran Amazon EC2.

Topik

- Luncurkan instans Amazon EC2 menggunakan template peluncuran
- Meluncurkan instans di grup Amazon EC2 Auto Scaling menggunakan template peluncuran
- Luncurkan Armada EC2 menggunakan template peluncuran
- Luncurkan Spot Fleet menggunakan template peluncuran

## Luncurkan instans Amazon EC2 menggunakan template peluncuran

Anda dapat menggunakan parameter yang terdapat dalam template peluncuran untuk meluncurkan instans Amazon EC2. Setelah memilih template peluncuran, tetapi sebelum meluncurkan instance, Anda dapat memodifikasi parameter peluncuran.

Instans yang diluncurkan menggunakan templat peluncuran secara otomatis diberi dua tanda dengan kunci `aws:ec2launchtemplate:id` dan `aws:ec2launchtemplate:version`. Anda tidak dapat menghapus atau mengedit tag ini.

Console

Untuk meluncurkan instance menggunakan template peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Gunakan salah satu opsi berikut untuk memilih template peluncuran:
   - Dari dasbor konsol Amazon EC2, pilih panah bawah di samping Luncurkan instance, pilih Luncurkan instance dari template, lalu untuk Template Sumber, pilih template peluncuran.
   - Di panel navigasi, pilih Luncur kan Template, pilih template peluncuran, dan pilih Tind akan, Luncurkan instance dari template.
3. Untuk Versi templat sumber, pilih versi templat peluncuran yang akan digunakan.
4. (Opsional) Anda dapat mengubah nilai untuk salah satu parameter peluncuran. Jika Anda tidak mengubah nilai, nilai yang ditentukan oleh template peluncuran digunakan. Jika tidak ada nilai yang ditentukan dalam template peluncuran, nilai default untuk parameter digunakan.
5. Di panel Ringkasan, untuk Jumlah instance, tentukan jumlah instance yang akan diluncurkan.
6. Pilih Luncurkan instans.

Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

AWS CLI

Untuk meluncurkan instance dari template peluncuran

- Gunakan perintah run-instances dan tentukan parameter `--launch-template`. Secara opsional, tentukan versi templat peluncuran yang akan digunakan. Jika Anda tidak menentukan versinya, versi default akan digunakan.

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123,Version=1
```

- Untuk mengganti parameter templat peluncuran, tentukan parameter di perintah run-instances. Contoh berikut menggantikan tipe instans yang ditentukan di templat peluncuran (jika ada).

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --instance-type t2.small
```

- Jika Anda menentukan parameter bersarang yang merupakan bagian dari struktur kompleks, instans akan diluncurkan menggunakan struktur kompleks seperti yang ditentukan dalam templat peluncuran ditambah parameter bersarang tambahan yang Anda tentukan.

  Dalam contoh berikut, instans diluncurkan dengan tanda *Owner*=*TeamA* serta tanda lainnya yang ditentukan di templat peluncuran. Jika templat peluncuran sudah memiliki tanda dengan kunci *Owner*, nilainya akan diganti dengan *TeamA*.

```
aws ec2 run-instances \
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --tag-specifications "ResourceType=instance,Tags=[{Key=Owner,Value=TeamA}]"
```

  Dalam contoh berikut, instans diluncurkan dengan volume dengan nama perangkat */dev/xvdb* serta pemetaan perangkat blok lainnya yang ditentukan dalam template peluncuran. Jika templat peluncuran sudah memiliki volume yang ditentukan untuk */dev/xvdb*, nilainya akan diganti dengan nilai yang ditentukan.

```
aws ec2 run-instances \
```

```
    --launch-template LaunchTemplateId=lt-0abcd290751193123 \
    --block-device-mappings "DeviceName=/dev/
xvdb,Ebs={VolumeSize=20,VolumeType=gp2}"
```

Jika instance gagal diluncurkan atau status langsung beralih ke `terminated` dari pada `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

PowerShell

Untuk meluncurkan instance dari template peluncuran menggunakan Alat AWS untuk PowerShell

- Gunakan perintah New-EC2Instance dan tentukan parameter `-LaunchTemplate`. Secara opsional, tentukan versi templat peluncuran yang akan digunakan. Jika Anda tidak menentukan versinya, versi default akan digunakan.

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -LaunchTemplate (
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
    }
)
```

- Untuk mengganti parameter template peluncuran, tentukan parameter dalam New-EC2Instance perintah. Contoh berikut menggantikan tipe instans yang ditentukan di templat peluncuran (jika ada).

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -InstanceType t4g.small `
    -LaunchTemplate (
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
    }
)
```

- Jika Anda menentukan parameter bersarang yang merupakan bagian dari struktur kompleks, instans akan diluncurkan menggunakan struktur kompleks seperti yang ditentukan dalam templat peluncuran ditambah parameter bersarang tambahan yang Anda tentukan.

  Dalam contoh berikut, instans diluncurkan dengan tanda *Owner*=*TeamA* serta tanda lainnya yang ditentukan di templat peluncuran. Jika templat peluncuran sudah memiliki tanda dengan kunci *Owner*, nilainya akan diganti dengan *TeamA*.

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -InstanceType t4g.small  `
    -LaunchTemplate (
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
        }
) `
    -TagSpecification (
        New-Object -TypeName Amazon.EC2.Model.TagSpecification -Property @{
            ResourceType = 'instance';
            Tags         = @(
                @{key = "Owner"; value = "TeamA" },
                @{key = "Department"; value = "Operations" }
            )
        }
)
```

  Dalam contoh berikut, instans diluncurkan dengan volume dengan nama perangkat */dev/xvdb* serta pemetaan perangkat blok lainnya yang ditentukan dalam template peluncuran. Jika templat peluncuran sudah memiliki volume yang ditentukan untuk */dev/xvdb*, nilainya akan diganti dengan nilai yang ditentukan.

```
Import-Module AWS.Tools.EC2
New-EC2Instance `
    -InstanceType t4g.small  `
    -LaunchTemplate (
        New-Object -TypeName Amazon.EC2.Model.LaunchTemplateSpecification -
Property @{
            LaunchTemplateId = 'lt-0abcd290751193123';
            Version          = '4'
```

```
    }
) `
    -BlockDeviceMapping  (
        New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping -Property @{
            DeviceName = '/dev/xvdb';
            EBS        = (
                New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice -Property @{
                    VolumeSize = 25;
                    VolumeType = 'gp3'
                }
            )
        }
)
```

Jika instance gagal diluncurkan atau status langsung beralih ke `terminated` dari pada `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

## Meluncurkan instans di grup Amazon EC2 Auto Scaling menggunakan template peluncuran

Anda dapat membuat grup Auto Scaling dan menentukan templat peluncuran yang akan digunakan untuk grup tersebut. Saat Amazon EC2 Auto Scaling meluncurkan instans di grup Auto Scaling, layanan ini menggunakan parameter peluncuran yang ditentukan di templat peluncuran terkait.

Sebelum Anda dapat membuat grup Penskalaan Otomatis menggunakan template peluncuran, Anda harus terlebih dahulu membuat template peluncuran yang menyertakan parameter yang diperlukan untuk meluncurkan instance dalam grup Penskalaan Otomatis. Beberapa parameter diperlukan, seperti ID AMI, dan beberapa parameter tidak tersedia untuk digunakan dengan grup Auto Scaling. Konsol menyediakan panduan untuk membantu Anda membuat template yang dapat Anda gunakan dengan Amazon EC2 Auto Scaling.

Untuk membuat grup Auto Scaling dengan template peluncuran menggunakan konsol

- Untuk petunjuknya, lihat Membuat grup Penskalaan Otomatis menggunakan template peluncuran di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

Untuk membuat atau memperbarui grup Auto Scaling dengan template peluncuran menggunakan AWS CLI

- Gunakan perintah create-auto-scaling-group atau update-auto-scaling-group dan tentukan parameter `--launch-template`.

Untuk informasi selengkapnya, lihat topik berikut di Panduan Pengguna Penskalaan Otomatis Amazon EC2:

- Buat templat peluncuran untuk grup Auto Scaling
- Buat template peluncuran menggunakan pengaturan lanjutan
- Contoh untuk membuat dan mengelola template peluncuran dengan AWS Command Line Interface (AWS CLI) - Menyediakan contoh yang menunjukkan cara membuat template peluncuran dengan berbagai kombinasi parameter.
- Buat grup Auto Scaling menggunakan template peluncuran
- Memperbarui grup Auto Scaling

## Luncurkan Armada EC2 menggunakan template peluncuran

Template peluncuran adalah persyaratan saat membuat permintaan Armada EC2. Saat Amazon EC2 memenuhi permintaan Armada EC2, layanan ini menggunakan parameter peluncuran yang ditentukan di templat peluncuran terkait. Anda dapat mengganti beberapa parameter yang ditentukan di templat peluncuran. Untuk informasi selengkapnya, lihat Membuat Armada EC2.

Untuk membuat Armada EC2 dengan template peluncuran menggunakan AWS CLI

- Gunakan perintah create-fleet. Gunakan parameter `--launch-template-configs` untuk menentukan templat peluncuran dan setiap penggantian untuk templat peluncuran.

## Luncurkan Spot Fleet menggunakan template peluncuran

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Template peluncuran bersifat opsional saat membuat permintaan Spot Fleet. Jika Anda tidak menggunakan template peluncuran, Anda dapat menentukan parameter peluncuran secara manual. Jika Anda menggunakan template peluncuran, saat Amazon EC2 memenuhi permintaan Spot Fleet, ia menggunakan parameter peluncuran yang ditentukan dalam template peluncuran terkait. Anda dapat mengganti beberapa parameter yang ditentukan di templat peluncuran. Untuk informasi selengkapnya, lihat Membuat Armada Spot.

Untuk membuat permintaan Spot Fleet menggunakan template peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih Minta Instans Spot.
4. Di bawah Parameter peluncuran, pilih Gunakan templat peluncuran.
5. Untuk Templat peluncuran, pilih templat peluncuran, dan kemudian, dari bidang ke kanan, pilih versi templat peluncuran.
6. Konfigurasikan Armada Spot Anda dengan memilih opsi yang berbeda di layar ini. Untuk informasi lebih lanjut tentang opsi, lihat Membuat permintaan Spot Fleet menggunakan parameter yang ditentukan.
7. Saat Anda siap membuat Armada Spot, pilih Luncurkan.

Untuk membuat permintaan Spot Fleet menggunakan template peluncuran

- Gunakan perintah request-spot-fleet. Gunakan parameter `LaunchTemplateConfigs` untuk menentukan templat peluncuran dan setiap penggantian untuk templat peluncuran.

## Luncurkan instance EC2 menggunakan detail dari instance yang ada

Konsol Amazon EC2 menyediakan opsi Lincurkan lebih banyak seperti ini yang memungkinkan Anda menggunakan instans saat ini sebagai dasar untuk meluncurkan instans lainnya. Opsi ini secara otomatis mengisi wizard peluncuran instans Amazon EC2 dengan detail konfigurasi tertentu dari instans yang dipilih.

Pertimbangan-pertimbangan

- Kami tidak mengklonakan instans Anda; kami hanya mereplikasi beberapa detail konfigurasi. Untuk membuat salinan instans Anda, pertama-tama buat AMI darinya, lalu luncurkan lebih banyak

instans dari AMI. Buat templat peluncuran untuk memastikan bahwa Anda meluncurkan instans menggunakan detail peluncuran yang sama.

- Instans saat ini harus berada dalam status `running`.

### Detail yang disalin

Detail konfigurasi berikut disalin dari instans yang dipilih dari wizard peluncuran instans:

- ID AMI
- Jenis instans
- Zona Ketersediaan, atau VPC dan subnet tempat instans yang dipilih berada
- Alamat IPv4 publik. Jika instans yang dipilih saat ini memiliki alamat IPv4 publik, instans baru menerima alamat IPv4 publik, terlepas dari pengaturan alamat IPv4 publik default instans yang dipilih. Untuk informasi selengkapnya tentang alamat IPv4 publik, lihat Alamat IPv4 publik.
- Grup penempatan, jika ada
- Peran IAM yang terkait dengan instans, jika berlaku
- Pengaturan perilaku pematian (berhenti atau berakhir)
- Pengaturan perlindungan pemutusan hubungan kerja (benar atau salah)
- CloudWatch pemantauan (diaktifkan atau dinonaktifkan)
- Peng EBS-optimization aturan Amazon (benar atau salah)
- Pengaturan penghunian, jika diluncurkan ke VPC (bersama atau khusus)
- ID Kernel dan ID disk RAM, jika ada
- Data pengguna, jika ditentukan
- Tanda yang terkait dengan instans, jika ada
- Grup keamanan yang terkait dengan instans
- [Instans Windows] Informasi asosiasi. Jika instans yang dipilih dikaitkan dengan file konfigurasi, file yang sama secara otomatis dikaitkan dengan instans baru. Jika file konfigurasi menyertakan konfigurasi domain gabungan, instans baru akan digabungkan ke domain yang sama. Untuk informasi selengkapnya tentang bergabung dengan domain, lihat Menggab ungkan instans Windows EC2 dengan lancar ke Direktori Aktif Microsoft AD Terkel AWS ola Anda di Panduan AWS Directory Service Administrasi.

Detail tidak disalin

Detail konfigurasi berikut tidak disalin dari instans yang Anda pilih. Sebaliknya, wizard menerapkan pengaturan atau perilaku default mereka:

- Jumlah antarmuka jaringan – Default-nya adalah satu antarmuka jaringan, yang merupakan antarmuka jaringan utama (eth0).
- Penyimpanan – Konfigurasi penyimpanan default ditentukan oleh AMI dan tipe instans.

Untuk meluncurkan lebih banyak instans seperti instans yang ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Tindakan, Gambar dan templat, Luncurkan lebih banyak yang seperti ini.
4. Wizard peluncuran instans akan terbuka. Anda dapat membuat perubahan yang diperlukan pada konfigurasi instans dengan memilih opsi yang berbeda di layar ini.

   Ketika Anda siap untuk meluncurkan instans Anda, pilih Luncurkan instans.
5. Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

## Luncurkan instans Amazon EC2 dari AMI AWS Marketplace

Anda dapat berlangganan AWS Marketplace AMI dan meluncurkan instance darinya menggunakan konsol Amazon EC2 atau alat baris perintah. Untuk informasi lebih lanjut tentang AWS Marketplace AMIs, lihatAMI berbayar di AWS Marketplace untuk instans Amazon EC2.

Untuk membatalkan langganan AMI setelah peluncuran, Anda harus terlebih dahulu menghentikan semua instance yang diluncurkan dari AMI. Untuk informasi selengkapnya, lihat Kelola Anda AWS Marketplace langganan.

Untuk meluncurkan instance dari AWS Marketplace AMI menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/
2. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.
3. (Opsional) Pada Nama dan tanda, untuk Nama, masukkan nama deskriptif untuk instans Anda.

4. Di bawah Application and OS Images (Amazon Machine Image) AMIs, pilih Browse more, lalu pilih AWS Marketplace AMIstab. Temukan AMI yang sesuai dengan menelusuri kategori, atau menggunakan fungsi pencarian. Untuk memilih produk, pilih Pilih.
5. Sebuah jendela terbuka dengan ikhtisar produk yang Anda pilih. Anda dapat melihat informasi harga, serta informasi lain yang disediakan vendor. Saat Anda siap, pilih Berlangganan dan luncurkan. Ini akan segera memulai langganan Anda. Saat berlangganan sedang berlangsung, Anda dapat mengonfigurasi instance dengan melanjutkan langkah-langkah dalam prosedur ini. Jika ada masalah dengan detail kartu kredit Anda, Anda akan diminta untuk memperbarui detail akun Anda.

   > **Note**
   >
   > Anda tidak dikenai biaya untuk menggunakan produk hingga Anda meluncurkan instans dengan AMI. Catat harga untuk setiap tipe instans yang didukung saat Anda memilih tipe instans. Pajak tambahan mungkin juga berlaku pada produk.

6. Untuk Tipe instans, pilih tipe instans untuk instans Anda. Tipe instans menentukan konfigurasi perangkat keras dan ukuran instans yang akan diluncurkan.
7. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang ada atau buat yang baru.
8. Di bawah Pengaturan Jaringan, untuk Firewall (grup keamanan), perhatikan grup keamanan baru yang dibuat sesuai dengan spesifikasi vendor untuk produk tersebut. Grup keamanan mungkin menyertakan aturan yang memungkinkan semua IPv4 alamat (`0.0.0.0/0`) akses pada SSH (port 22) di Linux atau RDP (port 3389) pada Windows. Kami menyarankan Anda menyesuaikan aturan ini untuk mengizinkan hanya alamat atau rentang alamat tertentu yang bisa mengakses instans Anda melalui port tersebut.
9. Anda dapat menggunakan bidang lain di layar untuk mengonfigurasi instans Anda, menambahkan penyimpanan, dan menambahkan tanda. Untuk informasi tentang berbagai opsi yang dapat Anda konfigurasikan, lihat Referensi untuk parameter konfigurasi EC2 instans Amazon.
10. Di panel Ringkasan, pada Gambar Perangkat Lunak (AMI), periksa detail AMI tempat Anda akan meluncurkan instans. Periksa juga detail konfigurasi lain yang Anda tentukan. Ketika Anda siap untuk meluncurkan instans Anda, pilih Launch instans.
11. Tergantung pada produk langganan Anda, instans mungkin memerlukan waktu beberapa menit atau lebih untuk diluncurkan. Jika ada masalah dengan detail kartu kredit Anda, Anda akan

diminta untuk memperbarui detail akun Anda. Saat halaman konfirmasi peluncuran ditampilkan, pilih Lihat semua instans untuk membuka halaman Instans.

> **Note**
>
> Anda akan dikenai harga langganan selama instans Anda dalam status `running`, meskipun sedang idle. Jika instans Anda dihentikan, Anda mungkin masih dikenai biaya untuk penyimpanan.

12. Saat instans Anda ada dalam status `running`, Anda dapat menyambungkannya. Untuk melakukan ini, pilih instans Anda di daftar, pilih Hubungkan, dan pilih opsi koneksi. Untuk informasi selengkapnya tentang terhubung ke instans Anda, lihat Sambungkan ke instans EC2.

> **Important**
>
> Periksa petunjuk penggunaan vendor dengan hati-hati, karena Anda mungkin perlu menggunakan nama pengguna tertentu untuk terhubung ke instans Anda. Untuk informasi tentang mengakses detail langganan Anda, lihat Kelola Anda AWS Marketplace langganan.

13. Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

Untuk meluncurkan instance dari AWS Marketplace AMI menggunakan alat baris perintah

Untuk meluncurkan instance dari AWS Marketplace produk menggunakan alat baris perintah, pertama-tama pastikan bahwa Anda berlangganan produk. Anda kemudian dapat meluncurkan sebuah instans dengan ID AMI produk menggunakan metode berikut:

| Metode | Dokumentasi |
| --- | --- |
| AWS CLI | Gunakan perintah run-instance, atau lihat topik berikut untuk informasi selengkapnya: Luncurkan instance Anda di Panduan Pengguna.AWS Command Line Interface |
| AWS Tools for Windows PowerShell | Gunakan New-EC2Instanceperintah, atau lihat topik berikut untuk informasi selengkapnya: Luncurkan Instans Amazon EC2 Menggunakan Windows PowerShell |

| Metode | Dokumentasi |
| --- | --- |
| API Kueri | MenggunakanRunInstances permintaan. |

## Sambungkan ke instans EC2

Instans Amazon EC2 Anda adalah server virtual di AWS Cloud. Untuk masuk ke instance Anda, Anda harus membuat koneksi ke instance. Cara Anda menyambung ke instance tergantung pada sistem operasi instans dan sistem operasi pada komputer yang Anda gunakan untuk menyambung ke instance. Tabel berikut merinci persyaratan untuk setiap metode koneksi.

| Opsi koneksi | Sistem operasi instans | Aturan lalu lintas masuk | Izin IAM | Peran profil instans | Perangkat lunak pada contoh | Perangkat lunak pada sistem penghubung | Pasangan kunci |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Klien SSH | Linux | Ya | Tidak | Tidak | Tidak | Ya | Ya |
| EC2 Instance Connect | Linux | Ya | Ya | Tidak | Ya [1] | Tidak | Tidak |
| PuTTY | Linux | Ya | Tidak | Tidak | Tidak | Ya | Ya |
| Klien RDP | Windows | Ya | Tidak | Tidak | Tidak | Ya | Ya [1] |
| Fleet Manager | Windows | Tidak | Ya | Ya | Ya [1] | Tidak | Ya |
| Manajer Sesi | Linux, Windows | Tidak | Ya | Ya | Ya [1] | Tidak | Tidak |
| Titik Akhir | Linux, Windows | Ya | Ya | Tidak | Tidak | Tidak | Tidak [3] |

| Opsi koneksi | Sistem operasi instans | Aturan lalu lintas masuk | Izin IAM | Peran profil instans | Perangkat lunak pada contoh | Perangkat lunak pada sistem penghubung | Pasangan kunci |
|---|---|---|---|---|---|---|---|
| EC2 Instance Connect | | | | | | | |

[1] Perangkat lunak yang diperlukan hanya pra-instal pada AMI tertentu. Anda dapat menginstal perangkat lunak yang diperlukan secara manual sesuai kebutuhan pada sistem operasi yang didukung.

[2] Pasangan kunci hanya diperlukan jika Anda menggunakan kata sandi yang dibuat secara acak untuk akun pengguna Administrator lokal.

[3] Pasangan kunci diperlukan jika Anda menggunakan metode koneksi SSH.

Untuk informasi selengkapnya, lihat dokumentasi untuk opsi koneksi yang ingin Anda gunakan.

Opsi koneksi

- Hubungkan ke instans Linux Anda menggunakan klien SSH
- Hubungkan ke instance Linux Anda menggunakan PuTTY
- Sambungkan ke instance Windows Anda menggunakan klien RDP
- Hubungkan ke instans Windows Anda menggunakan Fleet Manager
- Terhubung menggunakan Session Manager
- Hubungkan menggunakan IP publik dan EC2 Instance Connect
- Menghubungkan menggunakan IP pribadi dan EC2 Instance Connect Endpoint

## Prasyarat koneksi umum

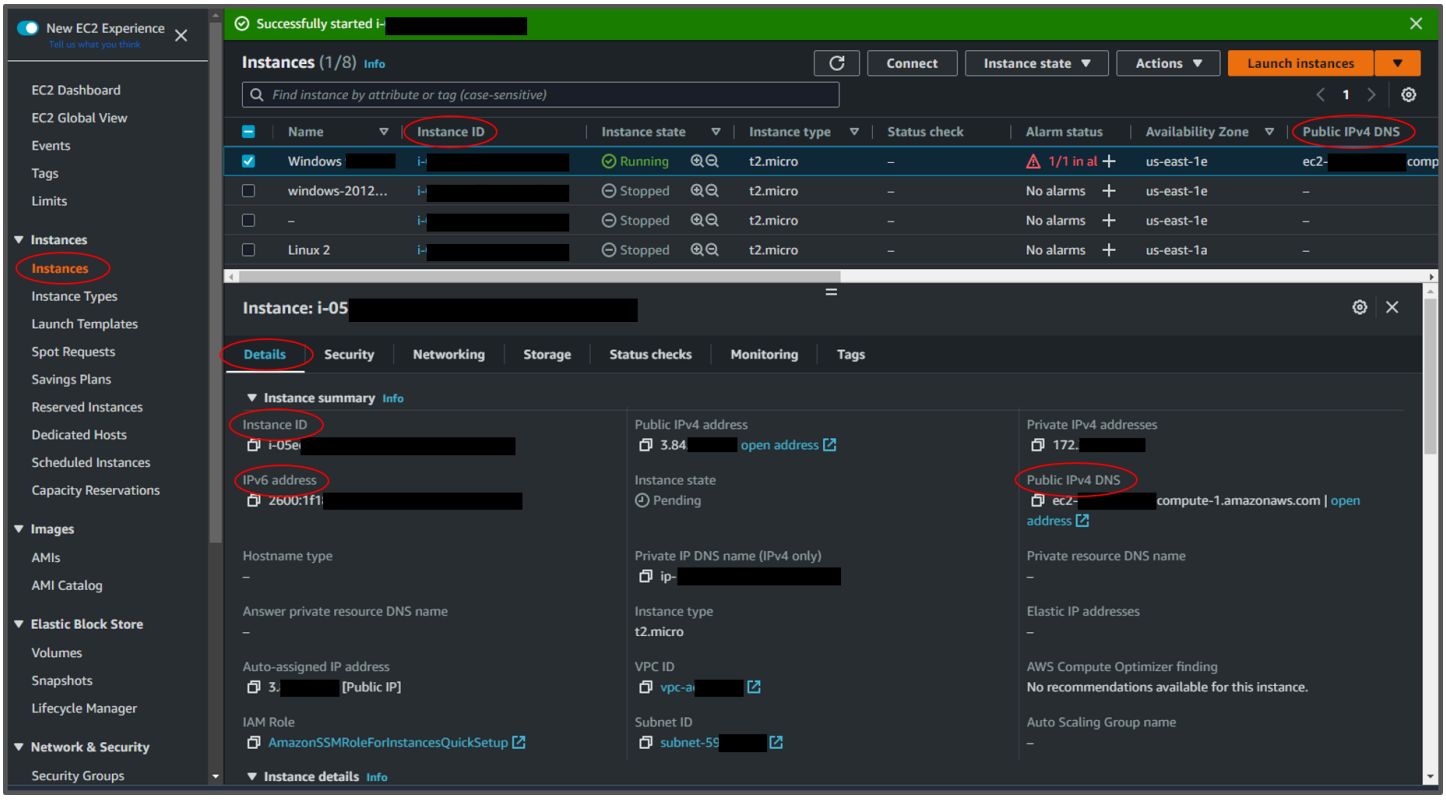
Berikut ini adalah prasyarat umum untuk terhubung ke instance. Perhatikan bahwa mungkin ada prasyarat tambahan yang khusus untuk opsi koneksi yang Anda pilih.

Prasyarat umum

- Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Diperlukan beberapa menit agar instance siap menerima permintaan koneksi. Untuk informasi selengkapnya, lihat Melihat pemeriksaan status.
- Dapatkan detail instans yang diperlukan.
- Temukan kunci pribadi dan atur izin.
- (Opsional) Dapatkan sidik jari instans.

## Dapatkan detail instans yang diperlukan

Untuk mempersiapkan koneksi ke instans Anda, dapatkan informasi berikut dari konsol Amazon EC2 atau dengan menggunakan baris perintah.




- Dapatkan nama DNS publik dari instans tersebut.

  Anda dapat memperoleh alamat DNS publik untuk instans Anda dari konsol Amazon EC2. Periksa kolom DNS IPv4 Publik pada panel instans. Jika kolom ini tersembunyi, pilih ikon pengaturan ( ) di sudut kanan atas layar dan pilih DNS Publik (IPv4). Anda juga dapat menemukan DNS publik di bagian

informasi instans pada panel Instans. Saat Anda memilih instans di panel Instans konsol Amazon EC2, informasi tentang instans tersebut akan muncul di bagian bawah halaman. Pada tab Detail, cari DNS IPv4 Publik.

Jika Anda mau, Anda dapat menggunakan perintah describe-instance (AWS CLI) atau Get-EC2Instance (AWS Tools for Windows PowerShell).

Jika tidak ada DNS IPv4 Publik yang ditampilkan, verifikasi bahwa status Instans sedang Berjalan, dan Anda belum meluncurkan instans di subnet pribadi. Jika Anda meluncurkan instans menggunakan wizard instance peluncuran, Anda mungkin telah mengedit bidang IP Auto-assign publik di bawah Pengaturan jaringan dan mengubah nilainya menjadi Non aktifkan. Jika Anda menonaktifkan opsi IP Auto-assign publik, instance tidak diberi alamat IP publik saat diluncurkan.

- (Hanya contoh IPv6) Dapatkan alamat IPv6 dari instance.

  Jika Anda telah menetapkan alamat IPv6 untuk instans Anda, maka Anda dapat secara opsional ter-connect ke instans menggunakan alamat IPv6-nya, bukan alamat IPv4 publik atau nama host DNS IPv4 publik. Komputer lokal Anda harus mempunyai alamat IPv6 dan harus dikonfigurasi untuk menggunakan IPv6. Anda dapat memperoleh alamat IPv6 instans Anda dari konsol Amazon EC2. Periksa kolom IP IPv6 pada panel instans. Atau, Anda dapat menemukan alamat IPv6 di bagian informasi instans. Saat Anda memilih instans di panel Instans konsol Amazon EC2, informasi tentang instans tersebut akan muncul di bagian bawah halaman. Pada tab Detail, cari alamat IPv6.

  Jika Anda mau, Anda dapat menggunakan perintah describe-instance (AWS CLI) atau Get-EC2Instance (AWS Tools for Windows PowerShell). Untuk informasi selengkapnya tentang IPv6, lihat Alamat IPv6.

- (Instans Linux) Dapatkan nama pengguna untuk instance Anda.

  Anda dapat terhubung ke instans menggunakan nama pengguna untuk akun pengguna Anda atau nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instans Anda.

  - Dapatkan nama pengguna untuk akun pengguna Anda.

    Untuk informasi selengkapnya tentang cara membuat akun pengguna, lihat Mengelola pengguna sistem di instans Amazon EC2 Linux Anda.

  - Dapatkan nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instans Anda.

    - Amazon Linux — `ec2-user`

- CentOS — atau `centos` `ec2-user`
- Debian — `admin`
- Fedora — atau `fedora` `ec2-user`
- FreeBSD — `ec2-user`
- RHEL — atau `ec2-user` `root`
- SUSE — atau `ec2-user` `root`
- Ubuntu — `ubuntu`
- Oracle – `ec2-user`
- Bitnamy — `bitnami`
- Rocky Linux — `rocky`
- Lain nya — Periksa dengan penyedia AMI

## Temukan kunci pribadi dan atur izin

Anda harus mengetahui lokasi file kunci pribadi Anda untuk membuat koneksi awal ke instance Linux menggunakan SSH atau instance Windows menggunakan RDP. Untuk koneksi SSH, Anda harus mengatur izin file sehingga hanya Anda yang dapat membaca kunci pribadi.

Untuk informasi tentang cara kerja pasangan kunci saat menggunakan Amazon EC2, lihat Pasangan kunci Amazon EC2 dan instans Amazon EC2.

- Temukan kunci pribadi.

  Dapatkan jalur yang memenuhi semua syarat ke lokasi di komputer Anda dari file `.pem` untuk pasangan kunci yang Anda tentukan saat meluncurkan instans. Untuk informasi selengkapnya, lihat the section called “Mengidentifikasi kunci publik yang ditentukan saat peluncuran”.

  Jika Anda tidak dapat menemukan file kunci pribadi Anda, lihat Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance saya?

  (Instans Linux) Jika Anda menghubungkan ke instance Anda menggunakan PuTTY dan perlu mengonversi `.pem` file ke`.ppk`, lihatKonversi kunci privat Anda menggunakan PuTTYgen.
- (Instans Linux) Tetapkan izin kunci pribadi Anda sehingga hanya Anda yang dapat membacanya.
  - Hubungkan dari macOS atau Linux

Jika Anda berencana menggunakan klien SSH pada komputer macOS atau Linux untuk terhubung ke instans Linux, gunakan perintah berikut untuk mengatur izin file kunci privat Anda sehingga hanya Anda yang dapat membacanya.

```
chmod 400 key-pair-name.pem
```

Jika Anda tidak mengatur izin tersebut, Anda tidak akan dapat terhubung ke instans Anda menggunakan pasangan kunci ini. Untuk informasi selengkapnya, lihat Kesalahan: File kunci privat yang tidak dilindungi.

- Hubungkan dari Windows

  Buka File Explorer dan buka menu konteks (klik kanan) untuk `.pem` file tersebut. Pilih Properties > Security tab dan pilih Advanced. Pilih Nonaktifkan warisan. Hapus akses ke semua pengguna kecuali untuk pengguna saat ini.

## (Opsional) Dapatkan sidik jari instans

Untuk melindungi diri dari serangan man-in-the-middle, Anda dapat memverifikasi keaslian instance yang akan Anda sambungkan dengan memverifikasi sidik jari yang ditampilkan. Memverifikasi sidik jari berguna jika Anda meluncurkan instans dari AMI publik yang disediakan oleh pihak ketiga.

Gambaran umum tugas

Pertama, dapatkan sidik jari instance dari instance. Kemudian, ketika Anda terhubung ke instance dan diminta untuk memverifikasi sidik jari, bandingkan sidik jari yang Anda peroleh dalam prosedur ini dengan sidik jari yang ditampilkan. Jika sidik jari tidak cocok, seseorang mungkin mencoba melakukan serangan man-in-the-middle. Jika cocok, Anda dapat dengan percaya diri terhubung ke instans Anda.

Prasyarat untuk mendapatkan sidik jari instans

- Instans tidak boleh dalam status `pending`. Sidik jari hanya tersedia setelah boot pertama instans selesai.
- Anda harus menjadi pemilik instans untuk mendapatkan output konsol.
- Ada berbagai cara untuk mendapatkan sidik jari instance. Jika Anda ingin menggunakan AWS CLI, itu harus diinstal pada komputer lokal Anda. Untuk informasi tentang menginstal AWS CLI, lihat Mem ulai dengan AWS CLI di Panduan AWS Command Line Interface Pengguna.

Untuk mendapatkan sidik jari instans

Pada Langkah 1, Anda mendapatkan output konsol, yang mencakup sidik jari instance. Pada Langkah 2, Anda menemukan sidik jari instance di output konsol.

1. Dapatkan output konsol menggunakan salah satu metode berikut.

   Console

   1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
   2. Dari navigator kiri, pilih Instans.
   3. Pilih instance Anda, lalu pilih Tindakan , Pan tau dan pemecahan masalah, Dapatkan log sistem.

   AWS CLI

   Di komputer lokal Anda (bukan pada instance yang Anda sambungkan), gunakan perintah get-consol e-output. Jika outputnya besar, Anda dapat menyalurkannya ke file teks, yang mungkin lebih mudah dibaca.

   ```
   aws ec2 get-console-output \
       --instance-id i-1234567890abcdef0 \
       --query Output \
       --output text > temp.txt
   ```

   PowerShell

   Di komputer lokal Anda, gunakan Get-EC2ConsoleOutput cmdlet berikut.

   ```
   $encodedOutput = (Get-EC2ConsoleOutput -InstanceId i-1234567890abcdef0).Output
   [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($encodedOutput
   ```

2. Di output konsol, temukan sidik jari instance (host), yang terletak di bawah`BEGIN SSH HOST KEY FINGERPRINTS`. Mungkin ada beberapa contoh sidik jari. Saat Anda terhubung ke instance Anda, itu hanya akan menampilkan salah satu sidik jari.

   Output yang tepat dapat bervariasi menurut sistem operasi, versi AMI, dan apakah AWS dibuat pasangan kunci. Berikut ini adalah output contoh.

   ```
   ec2:#############################################################
   ```

```
ec2: -----BEGIN SSH HOST KEY FINGERPRINTS-----
ec2: 256 SHA256:l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY no comment (ECDSA)
ec2: 256 SHA256:kpEa+rw/Uq3zxaYZN8KT501iBtJOIdHG52dFi66EEfQ no comment (ED25519)
ec2: 2048 SHA256:L8l6pepcA7iqW/jBecQjVZClUrKY+o2cHLI0iHerbVc no comment (RSA)
ec2: -----END SSH HOST KEY FINGERPRINTS-----
ec2: #############################################################
```

> **Note**
>
> Anda akan mereferensikan sidik jari ini saat terhubung ke instance.

## Hubungkan ke instans Linux Anda, menggunakan SSH.

Ada beberapa cara untuk terhubung ke instance Linux Anda menggunakan SSH. Beberapa cara tergantung pada sistem operasi komputer lokal tempat Anda terhubung. Metode lain berbasis browser, seperti EC2 Instance Connect atau AWS Systems Manager Session Manager, dan dapat digunakan dari komputer mana pun. Anda dapat menggunakan SSH untuk terhubung ke instance Linux Anda dan menjalankan perintah, atau menggunakan SSH untuk mentransfer file antara komputer lokal dan instance Anda.

Sebelum Anda menyambung ke instance Linux menggunakan SSH, lengkapi prasyarat berikut:

- Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Diperlukan beberapa menit agar instance siap menerima permintaan koneksi. Untuk informasi selengkapnya, lihat Melihat pemeriksaan status.
- Pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas SSH masuk dari alamat IP Anda. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.
- Dapatkan detail instans yang diperlukan.
- Temukan kunci pribadi dan atur izin.
- (Opsional) Dapatkan sidik jari instans.

Kemudian, pilih salah satu opsi berikut untuk terhubung ke instance Linux Anda menggunakan SSH.

- Hubungkan menggunakan klien SSH
- Menghubungkan Menggunakan PuTTY

- Mentransfer file menggunakan SCP

Jika Anda tidak dapat terhubung ke instans dan memerlukan bantuan pemecahan masalah, lihatMemecahkan masalah saat menyambung ke instans Amazon EC2 Linux.

## Hubungkan ke instans Linux Anda menggunakan klien SSH

Anda dapat menggunakan Secure Shell (SSH) untuk terhubung ke instance Linux Anda dari komputer lokal Anda. Untuk informasi selengkapnya tentang opsi lain, lihatSambungkan ke instans EC2.

> **Note**
>
> Jika Anda menerima kesalahan saat mencoba terhubung ke instans Anda, pastikan instans Anda memenuhi semua. Prasyarat koneksi SSH Jika memenuhi semua prasyarat, dan Anda masih tidak dapat terhubung ke instans Linux Anda, lihat Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux.

Daftar Isi


- Prasyarat koneksi SSH
- Hubungkan ke instans Linux Anda menggunakan klien SSH


### Prasyarat koneksi SSH

Sebelum Anda dapat terhubung ke instance Linux menggunakan SSH, selesaikan tugas berikut.

Lengkapi prasyarat umum.

- Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Diperlukan beberapa menit agar instance siap menerima permintaan koneksi. Untuk informasi selengkapnya, lihat Melihat pemeriksaan status.
- Dapatkan detail instans yang diperlukan.
- Temukan kunci pribadi dan atur izin.
- (Opsional) Dapatkan sidik jari instans.

Izinkan lalu lintas SSH masuk dari alamat IP Anda.

Pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas SSH masuk dari alamat IP Anda. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.

Instal klien SSH di komputer lokal Anda (jika diperlukan).

Komputer lokal Anda kemungkinan telah menginstal klien SSH secara default. Anda dapat memverifikasi ini dengan memasukkan perintah berikut di jendela terminal. Jika komputer Anda tidak mengenali perintah, Anda harus menginstal klien SSH.

```
ssh
```

Berikut ini adalah beberapa opsi yang mungkin untuk Windows. Jika komputer Anda menjalankan sistem operasi yang berbeda, lihat dokumentasi untuk sistem operasi tersebut untuk opsi klien SSH.

#### Instal OpenSSH di Windows

Setelah Anda menginstal OpenSSH di Windows, Anda dapat terhubung ke instance Linux Anda dari komputer Windows Anda menggunakan SSH. Sebelum Anda mulai, pastikan Anda memenuhi persyaratan berikut.

Versi Windows

Versi Windows di komputer Anda harus Windows Server 2019 atau yang lebih baru.

Untuk versi Windows sebelumnya, unduh dan instal sebagai Win32-OpenSSH gantinya.

PowerShell persyaratan

Untuk menginstal OpenSSH pada OS Windows Anda menggunakan PowerShell, Anda harus menjalankan PowerShell versi 5.1 atau yang lebih baru, dan akun Anda harus menjadi anggota grup Administrator bawaan. Jalan `$PSVersionTable.PSVersion` kan dari PowerShell untuk memeriksa PowerShell versi Anda.

Untuk memeriksa apakah Anda anggota grup Administrator bawaan, jalankan PowerShell perintah berikut:

```
(New-Object
 Security.Principal.WindowsPrincipal([Security.Principal.WindowsIdentity]::GetCurrent())).Is
```

Jika Anda adalah anggota grup Administrator bawaan, output-nya adalah `True`.

Untuk menginstal OpenSSH untuk Windows menggunakan PowerShell, jalankan PowerShell perintah berikut.

```
Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

Berikut ini adalah output contoh.

```
Path          :
Online        : True
RestartNeeded : False
```

Untuk menghapus OpenSSH dari Windows menggunakan PowerShell, jalankan PowerShell perintah berikut.

```
Remove-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0
```

Berikut ini adalah output contoh.

```
Path          :
Online        : True
RestartNeeded : True
```

Instal Subsistem Windows untuk Linux (WSL)

Setelah Anda menginstal WSL di Windows, Anda dapat terhubung ke instance Linux Anda dari komputer Windows Anda menggunakan alat baris perintah Linux, seperti klien SSH.

Ikuti petunjuk dalam Instal Subsistem Windows untuk Linux pada instance EC2 Windows Anda. Jika Anda mengikuti instruksi dalam panduan instalasi Microsoft, mereka menginstal distribusi Ubuntu Linux. Anda dapat menginstal distribusi Linux yang berbeda jika Anda mau.

Di jendela terminal WSL, salin `.pem` file (untuk pasangan kunci yang Anda tentukan untuk instance Anda saat peluncuran) dari Windows ke WSL. Perhatikan jalur yang sepenuhnya memenuhi syarat ke file `.pem` di WSL untuk digunakan saat menghubungkan ke instans Anda. Untuk informasi tentang cara menentukan jalur ke hard drive Windows Anda, lihat Bagaimana cara mengakses drive C saya?.

```
cp /mnt/<Windows drive letter>/path/my-key-pair.pem ~/WSL-path/my-key-pair.pem
```

Untuk informasi tentang membongkar Subsistem Windows untuk Linux, lihat Bagaimana cara membongkar Distribusi WSL?.

Hubungkan ke instans Linux Anda menggunakan klien SSH

Gunakan prosedur berikut untuk terhubung ke instans Linux Anda menggunakan klien SSH.

Untuk menyambung ke instance Anda menggunakan klien SSH

1. Buka jendela terminal di komputer Anda.
2. Gunakan ssh perintah untuk terhubung ke instance. Anda memerlukan detail tentang instance yang Anda kumpulkan sebagai bagian dari prasyarat. Misalnya, Anda memerlukan lokasi kunci pribadi (`.pem`file), nama pengguna, dan nama DNS publik atau alamat IPv6. Berikut ini adalah contoh perintah.
   - (DNS Publik) Untuk menggunakan nama DNS publik, masukkan perintah berikut.

     ```
     ssh -i /path/key-pair-name.pem instance-user-name@instance-public-dns-name
     ```
   - (IPv6) Atau, jika instance Anda memiliki alamat IPv6, masukkan perintah berikut untuk menggunakan alamat IPv6.

     ```
     ssh -i /path/key-pair-name.pem instance-user-name@2001:db8::1234:5678:1.2.3.4
     ```

   Berikut ini adalah contoh respons.

   ```
   The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (198-51-100-1)'
    can't be established.
   ECDSA key fingerprint is l4UB/neBad9tvkgJf1QZWxheQmR59WgrgzEimCG6kZY.
   Are you sure you want to continue connecting (yes/no)?
   ```
3. (Opsional) Pastikan sidik jari dalam peringatan keamanan cocok dengan sidik jari. Jika sidik jari ini tidak cocok, seseorang mungkin mencoba melakukan serangan "man-in-the-middle". Jika cocok, lanjutkan ke langkah berikutnya. Untuk informasi selengkapnya, lihat Mend apatkan sidik jari instans.
4. Masukkan **`yes`**.

   Anda akan melihat tanggapan seperti berikut:

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (ECDSA) to
 the list of known hosts.
```

## Hubungkan ke instance Linux Anda menggunakan PuTTY

Anda dapat terhubung ke instance Linux Anda menggunakan PuTTY, klien SSH gratis untuk Windows.

Jika Anda menjalankan Windows Server 2019 atau yang lebih baru, sebaiknya gunakan OpenSSH, alat konektivitas sumber terbuka untuk login jarak jauh menggunakan protokol SSH.

> **Note**
>
> Jika Anda menerima kesalahan saat mencoba terhubung ke instans Anda, pastikan instans Anda memenuhi semua. Prasyarat koneksi SSH Jika memenuhi semua prasyarat, dan Anda masih tidak dapat terhubung ke instans Linux Anda, lihat Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux.

Daftar Isi


- Prasyarat
- (Opsional) Konversi kunci pribadi Anda menggunakan PuttyGen
- Hubungkan ke instans Linux Anda


### Prasyarat

Sebelum Anda terhubung ke instance Linux menggunakan PuTTY, selesaikan tugas berikut.

Lengkapi prasyarat umum.

- Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Diperlukan beberapa menit agar instance siap menerima permintaan koneksi. Untuk informasi selengkapnya, lihat Melihat pemeriksaan status.
- Dapatkan detail instans yang diperlukan.
- Temukan kunci pribadi dan atur izin.
- (Opsional) Dapatkan sidik jari instans.

Izinkan lalu lintas SSH masuk dari alamat IP Anda.

Pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas SSH masuk dari alamat IP Anda. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.

Instal PuTTY di komputer lokal Anda (jika diperlukan).

Unduh dan pasang PuTTY dari halaman unduhan PuTTY. Jika Anda sudah memiliki versi PuTTY sebelumnya yang terpasang, kami merekomendasikan Anda mengunduh versi terbaru. Pastikan untuk menginstal seluruh rangkaian.

Konversikan kunci pribadi Anda ke format PPK menggunakan PuttyGen.

Anda harus menentukan kunci pribadi untuk pasangan kunci yang Anda tentukan saat meluncurkan instance. Jika Anda membuat kunci pribadi dalam format.pem, Anda harus mengubahnya menjadi file PPK untuk digunakan dengan PuTTY. Cari kunci pribadi (file.pem), lalu ikuti langkah-langkah diKonversi kunci privat Anda menggunakan PuTTYgen.

(Opsional) Konversi kunci pribadi Anda menggunakan PuttyGen

PuTTY tidak secara native mendukung format PEM untuk kunci SSH. PuTTY menyediakan alat bernama PuTTYgen, yang mengubah kunci PEM ke format PPK yang diperlukan untuk PuTTY. Jika Anda membuat kunci menggunakan format PEM alih-alih format PPK, Anda harus mengonversi kunci pribadi Anda (file.pem) ke dalam format ini (file.ppk) untuk digunakan dengan PuTTY.

Untuk mengonversi kunci pribadi Anda dari format PEM ke PPK

1. Dari menu Mulai, pilih Semua Program, PuTTY, PuTTYgen.
2. Di bawah tipe kunci yang akan dibuat, pilih RSA. Jika versi PuttyGen Anda tidak menyertakan opsi ini, pilih SSH-2 RSA.




3. Pilih Muat. Secara default, PuTTYgen hanya menampilkan file dengan ekstensi `.ppk`. Untuk menemukan file `.pem`, pilih opsi untuk menampilkan file dari semua tipe.



4. Pilih file `.pem` Anda untuk pasangan kunci yang Anda tentukan saat Anda meluncurkan instans dan memilih Buka. PuTTYgen menampilkan pemberitahuan bahwa file `.pem` berhasil diimpor. Pilih OKE.
5. Untuk menyimpan kunci dalam format yang dapat digunakan PuTTY, pilih Simpan kunci privat. PuTTYgen menampilkan peringatan tentang menyimpan kunci tanpa frasa sandi. Pilih Ya.

> **Note**
>
> Frasa sandi pada kunci privat adalah lapisan perlindungan ekstra. Meskipun kunci pribadi Anda ditemukan, kunci tersebut tidak dapat digunakan tanpa frasa sandi. Kelemahan menggunakan frasa sandi adalah membuat otomatisasi lebih sulit karena diperlukan campur tangan manusia untuk masuk ke sebuah instans, atau untuk menyalin file ke sebuah instans.

6. Tentukan nama yang sama untuk kunci yang Anda gunakan untuk pasangan kunci (misalnya, `key-pair-name`) dan pilih Simpan. PuTTY secara otomatis menambahkan ekstensi file `.ppk`.

Kunci privat Anda sekarang dalam format yang benar untuk digunakan dengan PuTTY. Anda sekarang dapat terhubung ke instans Anda menggunakan klien SSH PuTTY.

Hubungkan ke instans Linux Anda

Gunakan prosedur berikut untuk terhubung ke instans Linux Anda menggunakan klien PuTTY. Anda memerlukan file `.ppk` yang Anda buat untuk kunci privat Anda. Untuk informasi selengkapnya, lihat (Opsional) Konversi kunci pribadi Anda menggunakan PuttyGen di bagian sebelumnya. Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux.

Versi uji terakhir - PuTTY .78

Untuk terhubung ke instans Anda menggunakan PuTTY

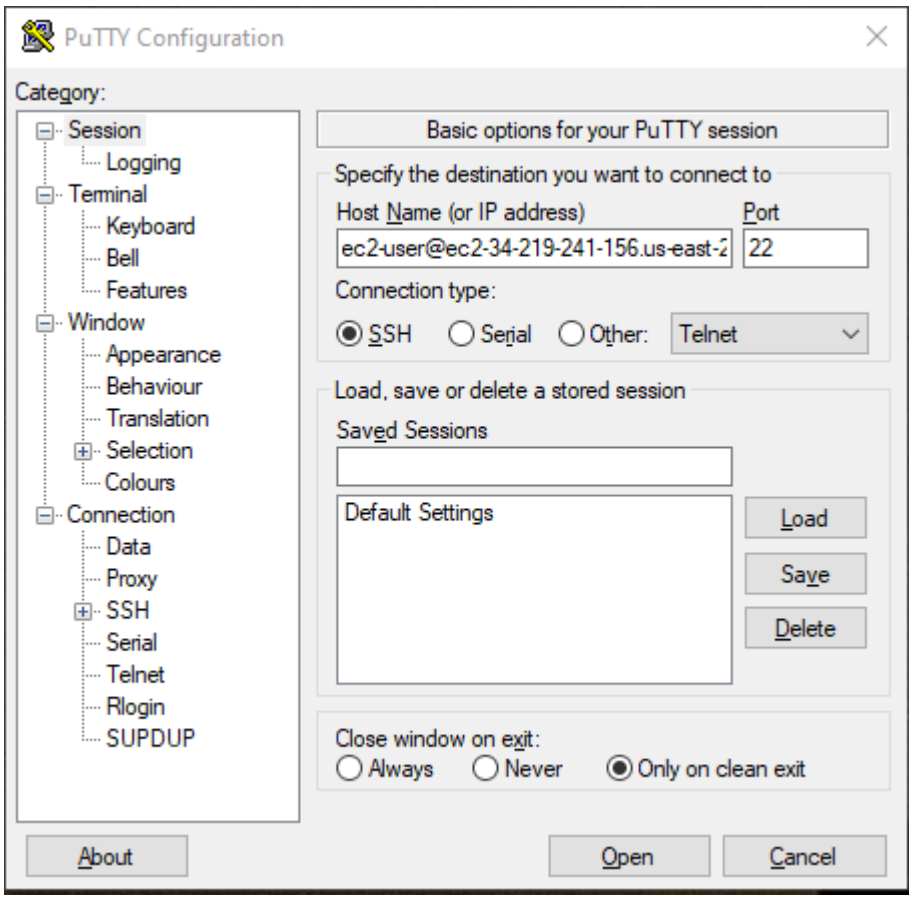
1. Mulai Putty (dari menu Mulai, cari Putty, kemudian pilih Buka).
2. Di panel Kategori, pilih Sesi dan lengkapi bidang berikut:
   a. Di kotak Nama Host, lakukan salah satu hal berikut ini:
      - (DNS Publik) Untuk menghubungkan menggunakan nama DNS publik instans Anda, masukkan *instance-user-name* @*instance-public-dns-name*.

- (IPv6) Atau, jika instans Anda memiliki alamat IPv6, untuk terhubung menggunakan alamat IPv6 instans Anda, masukkan @. *instance-user-name* *2001:db8::1234:5678:1.2.3.4*

Untuk informasi tentang cara mendapatkan nama pengguna untuk instans Anda, dan nama DNS publik atau alamat IPv6 instans Anda, lihat Dapatkan detail instans yang diperlukan.

b. Pastikan nilai Port adalah 22.

c. Di bawah Tipe koneksi, pilih SSH.




3. (Opsional) Anda dapat mengonfigurasi PuTTY untuk secara otomatis mengirim data 'keepalive' secara berkala agar sesi tetap aktif. Ini berguna untuk menghindari pemutusan hubungan dari instans Anda karena sesi tidak aktif. Di panel Kategori, pilih Koneksi, lalu masukkan interval yang diperlukan di Detik antara periode tetap aktif. Misalnya, jika sesi Anda terputus setelah 10 menit tidak ada aktivitas, masukkan 180 untuk mengonfigurasi PuTTY untuk mengirim data keepalive setiap 3 menit.
4. Di panel Kategori, perluas Koneksi, SSH, dan Otorisasi Pilih Kredensial.

5. Di samping File kunci pribadi untuk otentikasi, pilih Browse. Dalam kotak dialog Pilih file kunci privat, pilih file `.ppk` yang Anda buat untuk pasangan kunci Anda. Anda dapat membuka file atau memilih Buka di kotak dialog Pilih file kunci pribadi.
6. (Opsional) Jika Anda berencana untuk terhubung ke instans ini lagi setelah sesi ini, Anda dapat menyimpan informasi sesi untuk penggunaan di masa mendatang. Di panel Kategori, pilih Sesi. Masukkan nama untuk sesi di Sesi Tersimpan, lalu pilih Simpan.
7. Untuk terhubung ke instans, pilih Buka.
8. Jika ini adalah pertama kalinya Anda terhubung ke instans ini, PuTTY menampilkan kotak dialog peringatan keamanan yang menanyakan apakah Anda mempercayai host yang Anda sambungkan.

    a. (Opsional) Pastikan sidik jari di kotak dialog peringatan keamanan cocok dengan sidik jari yang Anda peroleh sebelumnya di (Opsional) Dapatkan sidik jari instans. Jika sidik jari ini tidak cocok, seseorang mungkin mencoba melakukan serangan "man-in-the-middle". Jika cocok, lanjutkan ke langkah berikutnya.

    b. Pilih Terima. Sebuah jendela terbuka dan Anda terhubung ke instans Anda.

> **Note**
>
> Jika Anda menentukan frasa sandi saat mengonversi kunci pribadi Anda ke format PuTTY, Anda harus memberikan frasa sandi tersebut saat masuk ke instans.

Jika Anda menemui kesalahan saat mencoba untuk terhubung ke instans, lihat Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux.

## Mentransfer file ke instance Linux menggunakan SCP

Salah satu cara untuk mentransfer file antara komputer lokal dan instans Linux Anda adalah dengan menggunakan protokol salinan aman (SCP). SCP adalah pilihan yang baik untuk operasi sederhana, seperti salinan file satu kali. SCP mengamankan transfer file menggunakan file.pem yang sama yang Anda gunakan untuk menyambung ke instance menggunakan SSH. Jika Anda perlu menyinkronkan file, atau jika file besar, rsync lebih cepat dan lebih efisien daripada SCP. Untuk keamanan, gunakan rsync melalui SSH, karena rsync mentransfer data menggunakan teks biasa secara default.

Sebelum Anda terhubung ke instance Linux menggunakan SCP, selesaikan tugas berikut:

- Lengkapi prasyarat umum.

  - Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Diperlukan beberapa menit agar instance siap menerima permintaan koneksi. Untuk informasi selengkapnya, lihat Melihat pemeriksaan status.
  - Dapatkan detail instans yang diperlukan.
  - Temukan kunci pribadi dan atur izin.
  - (Opsional) Dapatkan sidik jari instans.
- Izinkan lalu lintas SSH masuk dari alamat IP Anda.

  Pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas SSH masuk dari alamat IP Anda. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.
- Instal klien SCP.

  Kebanyakan komputer Linux, Unix, dan Apple menyertakan klien SCP secara default. Jika milik Anda tidak, proyek OpenSSH menyediakan implementasi gratis dari rangkaian lengkap alat SSH, termasuk klien SCP. Untuk informasi selengkapnya, lihat https://www.openssh.com.

Prosedur berikut ini akan menuntun Anda menggunakan SCP untuk mentransfer file menggunakan nama DNS publik instans, atau alamat IPv6 jika instans Anda memilikinya.

Untuk menggunakan SCP untuk mentransfer file antara komputer Anda dan instans Anda

1. Tentukan lokasi file sumber pada komputer Anda dan jalur tujuan pada instans. Dalam contoh berikut, nama file kunci pribadi adalah`key-pair-name.pem`, file yang akan ditransfer adalah`my-file.txt`, nama pengguna untuk instance adalahec2-user, nama DNS publik dari instance adalah`instance-public-dns-name`, dan alamat IPv6 instance adalah`2001:db8::1234:5678:1.2.3.4`.
   - (DNS Publik) Untuk mentransfer file ke tujuan yang ada di instans, masukkan perintah berikut dari komputer Anda.

     ```
     scp -i /path/key-pair-name.pem /path/my-file.txt ec2-user@instance-public-dns-name:path/
     ```
   - (IPv6) Untuk mentransfer file ke tujuan yang ada di instans jika instans memiliki alamat IPv6, masukkan perintah berikut dari komputer Anda. Alamat IPv6 harus diapit tanda kurung siku (`[ ]`), yang harus di-escape (\).

```
scp -i /path/key-pair-name.pem /path/my-file.txt ec2-user@
\[2001:db8::1234:5678:1.2.3.4\]:path/
```

2. Jika Anda belum ter-connect ke instans menggunakan SSH, maka Anda akan melihat respons seperti berikut:

```
The authenticity of host 'ec2-198-51-100-1.compute-1.amazonaws.com (10.254.142.33)'
can't be established.
RSA key fingerprint is 1f:51:ae:28:bf:89:e9:d8:1f:25:5d:37:2d:7d:b8:ca:9f:f5:f1:6f.
Are you sure you want to continue connecting (yes/no)?
```

   (Opsional) Anda dapat secara opsional memverifikasi bahwa sidik jari di peringatan keamanan cocok dengan sidik jari instans. Untuk informasi selengkapnya, lihat (Opsional) Dapatkan sidik jari instans.

   Masukkan **yes**.

3. Jika transfer berhasil, maka responsnya sama dengan berikut ini:

```
Warning: Permanently added 'ec2-198-51-100-1.compute-1.amazonaws.com' (RSA)
to the list of known hosts.
my-file.txt                                   100%   480     24.4KB/s   00:00
```

4. Untuk mentransfer sebuah file ke arah lain (dari instans Amazon EC2 ke komputer Anda), balikkan urutan parameter host. Misalnya, Anda dapat mentransfer `my-file.txt` dari instans EC2 Anda ke tujuan di komputer lokal Anda sebagai `my-file2.txt`, seperti yang ditunjukkan dalam contoh berikut.

   - (DNS Publik) Untuk mentransfer file ke tujuan yang ada di instans, masukkan perintah berikut dari komputer Anda.

     ```
     scp -i /path/key-pair-name.pem ec2-user@instance-public-dns-name:path/my-
     file.txt path/my-file2.txt
     ```

   - (IPv6) Untuk mentransfer file ke tujuan di komputer Anda jika instans memiliki alamat IPv6, masukkan perintah berikut dari komputer Anda. Alamat IPv6 harus diapit tanda kurung siku ([ ]), yang harus di-escape (\).

     ```
     scp -i /path/key-pair-name.pem ec2-user@\[2001:db8::1234:5678:1.2.3.4\]:path/
     my-file.txt path/my-file2.txt
     ```

# Mengelola pengguna sistem di instans Amazon EC2 Linux Anda

Setiap instans Linux diluncurkan dengan pengguna sistem Linux default. Anda dapat menambahkan pengguna ke instans Anda dan menghapus pengguna.

Untuk pengguna default, nama pengguna default ditentukan oleh AMI yang ditentukan saat Anda meluncurkan instance.

> **Note**
>
> Secara default, autentikasi kata sandi dan login root dinonaktifkan, dan sudo diaktifkan. Untuk masuk ke instans Anda, Anda harus menggunakan pasangan kunci. Untuk informasi selengkapnya tentang logging, lihat Hubungkan ke instans Linux Anda, menggunakan SSH.. Anda dapat mengizinkan autentikasi kata sandi dan login root untuk instans Anda. Untuk informasi selengkapnya tentang bagaimana menonaktifkan driver perangkat, lihat dokumentasi untuk sistem operasi Anda.

> **Note**
>
> Pengguna sistem Linux jangan bingung dengan pengguna (IAM). Untuk informasi selengkapnya, lihat Pengguna IAM di Panduan Pengguna IAM.

Daftar Isi


- Nama pengguna default
- Pertimbangan-pertimbangan
- Buat pengguna
- Menghapus pengguna


## Nama pengguna default

Nama pengguna default untuk instans EC2 ditentukan oleh AMI yang ditentukan saat Anda meluncurkan instans.

Nama pengguna default adalah:

- Untuk AMI Amazon Linux, nama penggunanya adalah`ec2-user`.
- Untuk CentOS AMI, nama penggunanya adalah `centos` atau`ec2-user`.
- Untuk AMI Debian, nama penggunanya adalah`admin`.
- Untuk Fedora AMI, nama penggunanya adalah `fedora` atau`ec2-user`.
- Untuk AMI FreeBSD, nama penggunanya adalah. `ec2-user`
- Untuk AMI RHEL, nama penggunanya adalah `ec2-user` atau`root`.
- Untuk SUSE AMI, nama penggunanya adalah `ec2-user` atau`root`.
- Untuk AMI Ubuntu, nama penggunanya adalah`ubuntu`.
- Untuk Oracle AMI, nama penggunanya adalah`ec2-user`.
- Untuk AMI Bitnami, nama penggunanya adalah`bitnami`.

> **Note**
>
> Untuk menemukan nama pengguna default untuk distribusi Linux lainnya, periksa dengan penyedia AMI.

Pertimbangan-pertimbangan

Menggunakan pengguna default sudah cukup untuk banyak aplikasi. Namun, Anda dapat memilih untuk menambahkan pengguna sehingga individu dapat memiliki file dan ruang kerja mereka sendiri. Lebih jauh lagi, membuat pengguna untuk pengguna baru jauh lebih aman daripada memberikan beberapa (mungkin tidak berpengalaman) akses pengguna default, karena pengguna default dapat menyebabkan banyak kerusakan pada sistem bila digunakan dengan tidak benar.

Untuk mengaktifkan akses SSH pengguna ke instans EC2 Anda menggunakan pengguna sistem Linux, Anda harus membagikan kunci SSH dengan pengguna. Atau, Anda dapat menggunakan EC2 Instance Connect untuk memberikan akses kepada pengguna tanpa perlu membagikan dan mengelola kunci SSH. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda menggunakan alamat IP publik dan EC2 Instance Connect.

Buat pengguna

Pertama buat pengguna, lalu tambahkan kunci publik SSH yang memungkinkan pengguna untuk terhubung dan masuk ke instans.

> **Important**
>
> Pada Langkah 1 dari prosedur ini, Anda membuat pasangan kunci baru. Karena pasangan kunci berfungsi seperti kata sandi, sangat penting untuk menanganinya dengan aman. Jika Anda membuat pasangan kunci untuk pengguna, Anda harus memastikan bahwa kunci pribadi dikirim kepada mereka dengan aman. Atau, pengguna dapat menyelesaikan Langkah 1 dan 2 dengan membuat pasangan kunci mereka sendiri, menjaga kunci pribadi tetap aman di mesin mereka, dan kemudian mengirimkan kunci publik untuk menyelesaikan prosedur dari Langkah 3.

Untuk membuat pengguna

1. Buat pasangan kunci baru. Anda harus menyediakan file `.pem` untuk pengguna yang Anda buat penggunanya. Mereka harus menggunakan file ini untuk terhubung ke instans.
2. Ambil kunci publik dari pasangan kunci yang Anda buat di langkah sebelumnya.

   ```
   $ ssh-keygen -y -f /path_to_key_pair/key-pair-name.pem
   ```

   Perintah tersebut mengembalikan kunci publik, seperti yang ditunjukkan pada contoh berikut.

   ```
   ssh-rsa
    AAAAB3NzaC1yc2EAAAADAQABAAABAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS7O6Vhz2ItxCih
   +PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/
   d6RJhJOI0iBXrlsLnBItntckiJ7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/
   i8SeJtjnV3iAoG/cQk+0FzZqaeJAAHco
   +CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi
   +z7wB3RbBQoQzd8v7yeb7OzlPnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
   ```

3. Hubungkan dengan instans.
4. Gunakan perintah adduser untuk membuat pengguna dan menambahkannya ke sistem (dengan entri di file `/etc/passwd`). Perintah tersebut juga membuat grup dan direktori home untuk pengguna. Dalam contoh ini, pengguna dinamai *newuser*.

   - AL2023 dan Amazon Linux 2

     Dengan AL2023 dan Amazon Linux 2, pengguna dibuat dengan otentikasi kata sandi dinonaktifkan secara default.

```
[ec2-user ~]$ sudo adduser newuser
```

- Ubuntu

  Sertakan `--disabled-password` parameter untuk membuat pengguna dengan otentikasi kata sandi dinonaktifkan.

```
[ubuntu ~]$ sudo adduser newuser --disabled-password
```

5. Beralih ke pengguna baru agar direktori dan file yang Anda buat memiliki kepemilikan yang sesuai.

```
[ec2-user ~]$ sudo su - newuser
```

   Perintah berubah dari `ec2-user` *`newuser`* ke untuk menunjukkan bahwa Anda telah mengalihkan sesi shell ke pengguna baru.

6. Tambahkan kunci publik SSH ke pengguna. Pertama buat direktori di direktori utama pengguna untuk file kunci SSH, lalu buat file kunci, dan terakhir tempelkan kunci publik ke dalam file kunci, seperti yang dijelaskan di sub-langkah berikut.

   a. Membuat direktori `.ssh` di direktori beranda *`newuser`* dan mengubah izin filenya menjadi `700` (hanya pemilik yang dapat membaca, menulis, atau membuka direktori).

```
[newuser ~]$ mkdir .ssh
```

```
[newuser ~]$ chmod 700 .ssh
```

> **Important**
>
> Tanpa izin file yang tepat ini, pengguna tidak akan bisa masuk.

   b. Membuat file bernama `authorized_keys` di direktori `.ssh` dan mengubah izin filenya menjadi `600` (hanya pemilik yang dapat membaca atau menulis ke file).

```
[newuser ~]$ touch .ssh/authorized_keys
```

```
[newuser ~]$ chmod 600 .ssh/authorized_keys
```

> **Important**
>
> Tanpa izin file yang tepat ini, pengguna tidak akan bisa masuk.

c. Buka file `authorized_keys` menggunakan editor teks favorit Anda (seperti vim atau nano).

```
[newuser ~]$ nano .ssh/authorized_keys
```

Tempel kunci publik yang Anda ambil pada Langkah 2 ke dalam file dan simpan perubahannya.

> **Important**
>
> Pastikan Anda menempelkan kunci publik dalam satu baris berkelanjutan. Kunci publik tidak boleh dibagi menjadi beberapa baris.

Pengguna sekarang harus dapat masuk ke *newuser* pengguna di instans Anda, menggunakan kunci privat yang sesuai dengan kunci publik yang Anda tambahkan ke `authorized_keys` file. Untuk informasi selengkapnya tentang berbagai metode untuk menghubungkan ke instans Linux, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..

### Menghapus pengguna

Jika pengguna tidak lagi diperlukan, Anda dapat menghapus pengguna tersebut sehingga tidak dapat digunakan lagi.

Gunakan perintah userdel untuk menghapus pengguna dari sistem. Saat Anda menentukan parameter `-r`, direktori beranda pengguna dan spool email akan dihapus. Untuk mempertahankan home directory dan mail spool pengguna, hilangkan parameter `-r`.

```
[ec2-user ~]$ sudo userdel -r olduser
```

# Hubungkan ke instans Windows Anda menggunakan RDP

Anda dapat terhubung ke instans Amazon EC2 yang dibuat dari sebagian besar Windows Amazon Machine Images (AMI) menggunakan Remote Desktop. Desktop Jarak Jauh menggunakan Remote Desktop Protocol (RDP) untuk terhubung ke dan gunakan instans Anda dengan cara yang sama seperti Anda menggunakan komputer yang ada di depan Anda (komputer lokal). Ini tersedia di sebagian besar edisi Windows dan juga tersedia untuk Mac OS.

Lisensi untuk sistem operasi Windows Server memungkinkan dua koneksi jarak jauh secara bersamaan untuk tujuan administratif. Lisensi untuk Windows Server sudah termasuk dalam harga instans Windows Anda. Jika Anda membutuhkan lebih dari dua koneksi jarak jauh secara bersamaan, Anda harus membeli lisensi Remote Desktop Services (RDS). Jika Anda mencoba koneksi ketiga, terjadi kesalahan.

> **Tip**
>
> Jika Anda perlu terhubung ke instance Anda untuk memecahkan masalah boot, konfigurasi jaringan, dan masalah lain untuk instans yang dibangun di Sistem AWS Nitro, Anda dapat menggunakan. EC2 Serial Console untuk instans

Daftar Isi


- Sambungkan ke instance Windows Anda menggunakan klien RDP
- Hubungkan ke instans Windows Anda menggunakan Fleet Manager
- Mentransfer file ke instance Windows menggunakan RDP


## Sambungkan ke instance Windows Anda menggunakan klien RDP

Anda dapat terhubung ke instance Windows Anda menggunakan klien RDP sebagai berikut.

> **Tip**
>
> Atau, Anda dapat menyambung ke instans Windows menggunakan Systems Manager Fleet Manager atau EC2 Instance Connect Endpoint.

### Prasyarat

Anda harus memenuhi prasyarat berikut untuk terhubung ke instance Windows Anda menggunakan klien RDP.

- Lengkapi prasyarat umum.
  - Periksa apakah pesan Anda telah lulus pemeriksaan statusnya. Diperlukan beberapa menit agar instance siap menerima permintaan koneksi. Untuk informasi selengkapnya, lihat Melihat pemeriksaan status.
  - Dapatkan detail instans yang diperlukan.
  - Temukan kunci pribadi dan atur izin.
  - (Opsional) Dapatkan sidik jari instans.
- Instal klien RDP.
  - (Windows) Windows menyertakan klien RDP secara default. Untuk memverifikasi, ketik mstsc di jendela Command Prompt. Jika komputer Anda tidak mengenali perintah ini, unduh aplikasi Microsoft Remote Desktop dari Microsoft Store.
  - (macOS X) Unduh Aplikasi Windows untuk Mac (sebelumnya bernama Microsoft Remote Desktop dari Mac App Store.
  - (Linux) Gunakan Remmina.
- Izinkan lalu lintas RDP masuk dari alamat IP Anda.

  Pastikan grup keamanan yang terkait dengan instance Anda mengizinkan lalu lintas RDP masuk dari alamat IP Anda. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.

### Ambil kata sandi administrator

Jika Anda menggabungkan instance ke domain, Anda dapat menyambung ke instans menggunakan kredentif domain dari Directory Service. Pada layar login Remote Desktop, alih-alih menggunakan nama komputer lokal dan kata sandi yang dihasilkan, gunakan nama pengguna yang sepenuhnya memenuhi syarat untuk administrator (misalnya,**`corp.example.com\Admin`**), dan kata sandi untuk akun ini.

Untuk menyambung ke instance Windows menggunakan RDP, Anda harus mengambil kata sandi administrator awal dan kemudian memasukkan kata sandi ini saat Anda terhubung ke instance Anda. Diperlukan beberapa menit setelah peluncuran instans sebelum sandi ini tersedia. Akun Anda harus

memiliki izin untuk melakukan GetPasswordData tindakan. Untuk informasi selengkapnya, lihat Contoh kebijakan untuk mengontrol akses Amazon EC2 API.

Nama pengguna default untuk akun Administrator tergantung pada bahasa sistem operasi (OS) yang terdapat dalam AMI. Untuk menentukan nama pengguna yang benar, identifikasi bahasa OS, lalu pilih nama pengguna yang sesuai. Misalnya, untuk OS Inggris, nama penggunanya adalah`Administrator`, untuk OS Prancis itu`Administrateur`, dan untuk OS Portugis itu`Administrador`. Jika versi bahasa OS tidak memiliki nama pengguna dalam bahasa yang sama, pilih nama penggguna`Administrator (Other)`. Untuk informasi selengkapnya, lihat Nama Lokalisasi untuk Akun Administrator di Windows di situs web Microsoft.

Untuk mengambil kata sandi administrator awal

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Hubungkan.
4. Pada halaman Connect to instance, pilih tab klien RDP.
5. Untuk Nama Pengguna, pilih nama pengguna default untuk akun Administrator. Nama pengguna yang Anda pilih harus sesuai dengan bahasa sistem operasi (OS) yang terdapat dalam AMI yang Anda gunakan untuk meluncurkan instans Anda. Jika tidak ada nama pengguna dalam bahasa yang sama dengan OS Anda, pilih Administrator (Lainnya).
6. Pilih Dapatkan kata sandi.
7. Pada halaman Dapatkan kata sandi Windows, lakukan hal berikut:

    a. Pilih Unggah file kunci pribadi dan arahkan ke file private key (`.pem`) yang Anda tentukan saat meluncurkan instance. Pilih file dan pilih Buka untuk menyalin seluruh isi file ke jendela ini.

    b. Pilih Dek ripsi kata sandi. Halaman Dapatkan kata sandi Windows ditutup, dan kata sandi administrator default untuk instance muncul di bawah Kata Sand i, menggantikan tautan Dapatkan kata sandi yang ditampilkan sebelumnya.

    c. Salin kata sandi dan simpan di tempat yang aman. Kata sandi ini diperlukan untuk terhubung ke instans.

Menghubungkan ke Instans Windows Anda

Prosedur berikut menggunakan klien Remote Desktop Connection untuk Windows (MSTSC). Jika Anda menggunakan klien RDP yang berbeda, unduh file RDP dan kemudian lihat dokumentasi untuk klien RDP untuk langkah-langkah untuk membuat koneksi RDP.

Untuk menyambung ke instance Windows menggunakan klien RDP

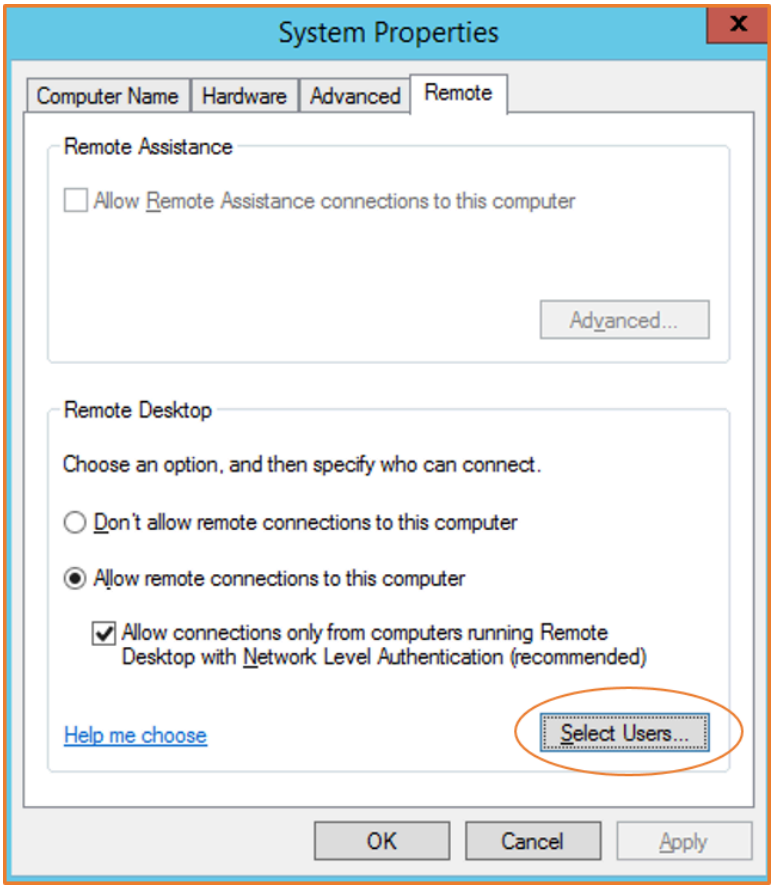
1. Pada halaman Hubungkan ke instance, pilih Unduh file desktop jarak jauh. Setelah pengunduhan file selesai, pilih Bat al untuk kembali ke halaman Instans. File RDP diunduh ke `Downloads` folder Anda.
2. Jalan `mstsc.exe` kan untuk membuka klien RDP.
3. Perluas Tampilkan opsi, pilih Buka, dan pilih file.rdp dari folder Anda`Downloads`.
4. Secara default, Komputer adalah nama DNS IPv4 publik dari instance dan Nama Pengguna adalah akun administrator. Untuk menyambung ke instance menggunakan IPv6 sebagai gantinya, ganti nama DNS IPv4 publik dari instance dengan alamat IPv6-nya. Tinjau pengaturan default dan ubah sesuai kebutuhan.
5. Pilih Hubungkan. Jika Anda menerima peringatan bahwa penerbit koneksi jarak jauh tidak diketahui, pilih Hubun gkan untuk melanjutkan.
6. Masukkan kata sandi yang Anda simpan sebelumnya, lalu pilih OK.
7. Karena sifat dari sertifikat yang ditandatangani sendiri, Anda mungkin mendapatkan peringatan bahwa sertifikat keamanan tidak dapat diautentikasi. Lakukan salah satu tindakan berikut:
   - Jika Anda mempercayai sertifikat, pilih Ya untuk menyambung ke instans Anda.
   - [Windows] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai di log sistem untuk mengkonfirmasi identitas komputer jarak jauh. Pilih Lihat sertifikat dan kemudian pilih Thumbprint dari tab Detail. Bandingkan nilai ini dengan nilai `RDPCERTIFICATE-THUMBPRINT` di Tindakan , Monitor dan pemecahan masalah, Dapatkan log sistem.
   - [Mac OS X] Sebelum Anda melanjutkan, bandingkan sidik jari sertifikat dengan nilai di log sistem untuk mengonfirmasi identitas komputer jarak jauh. Pilih Tampilkan Sertifikat, perluas Detail, dan pilih Sidik J ari SHA1. Bandingkan nilai ini dengan nilai `RDPCERTIFICATE-THUMBPRINT` di Tindakan , Monitor dan pemecahan masalah, Dapatkan log sistem.
8. Jika koneksi RDP berhasil, klien RDP menampilkan layar login Windows dan kemudian desktop Windows. Jika Anda menerima pesan kesalahan sebagai gantinya, lihatthe section called

"Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh". Ketika Anda selesai dengan koneksi RDP, Anda dapat menutup klien RDP.

### Konfigurasikan akun pengguna

Setelah tersambung ke instance melalui RDP, sebaiknya lakukan tugas berikut:

- Ubah kata sandi administrator dari nilai default. Anda dapat mengubah kata sandi saat masuk ke instans itu sendiri, seperti yang Anda lakukan di komputer mana pun yang menjalankan Windows Server.
- Buat pengguna lain dengan hak akses administrator di instans tersebut. Ini adalah perlindungan jika Anda lupa kata sandi administrator atau memiliki masalah dengan akun administrator. Pengguna baru harus memiliki izin untuk mengakses instans dari jarak jauh. Buka System Properties dengan mengklik kanan ikon This PC di desktop Windows atau File Explorer dan pilih Properties. Pilih Pengaturan jarak jauh, dan pilih Pilih Pengguna untuk menambahkan pengguna ke grup Pengguna Desktop Jarak Jauh.



## Hubungkan ke instans Windows Anda menggunakan Fleet Manager

Anda dapat menggunakan Fleet Manager, kemampuan AWS Systems Manager, untuk terhubung ke instans Windows menggunakan Remote Desktop Protocol (RDP) dan menampilkan hingga empat instans Windows pada halaman yang sama di Konsol Manajemen AWS. Anda dapat terhubung ke instans pertama di Desktop Jarak Jauh Manajer Armada langsung dari halaman Instans di konsol Amazon EC2. Untuk informasi selengkapnya tentang Fleet Manager, lihat Menghubungkan ke instance terkelola menggunakan Desktop J arak Jauh di Panduan AWS Systems Manager Pengguna.

Anda tidak perlu secara khusus mengizinkan lalu lintas RDP masuk dari alamat IP Anda jika Anda menggunakan Fleet Manager untuk terhubung. Manajer Armada menangani itu untuk Anda.

Prasyarat

Sebelum mencoba menyambung ke instance menggunakan Fleet Manager, Anda harus mengatur lingkungan Anda. Untuk informasi selengkapnya, lihat Men yiapkan lingkungan Anda di Panduan AWS Systems Manager Pengguna.

Untuk menyambung ke instance Windows menggunakan Fleet Manager

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Hubungkan.
4. Pada tab klien RDP, untuk Jenis Kon eksi, pilih Hubun gkan menggunakan Fleet Manager.
5. Pilih Desktop Jarak Jauh Fleet Manager. Halaman Desktop Jarak Jauh Fleet Manager akan terbuka di konsol AWS Systems Manager .
6. Masukkan kredensibilitas Anda, lalu pilih Hu bungkan.
7. Jika koneksi RDP berhasil, Fleet Manager menampilkan desktop Windows. Ketika Anda selesai dengan sesi, pilih Tindakan , Akhiri sesi.

Untuk informasi selengkapnya, lihat Menghubungkan ke instance terkelola Windows Server menggunakan Remote Desktop di Panduan AWS Systems Manager Pengguna.

## Mentransfer file ke instance Windows menggunakan RDP

Anda dapat bekerja dengan instans Windows Anda dengan cara yang sama seperti Anda bekerja dengan server Windows mana pun. Misalnya, Anda dapat mentransfer file antara instance Windows

dan komputer lokal Anda menggunakan fitur berbagi file lokal perangkat lunak Microsoft Remote Desktop Connection (RDP). Anda dapat mengakses file lokal di drive hard disk, drive DVD, drive media portabel, dan drive jaringan yang dipetakan.

Untuk mengakses file lokal Anda dari instans Windows, Anda harus mengaktifkan fitur berbagi file lokal dengan memetakan drive sesi jarak jauh ke drive lokal Anda. Langkah-langkahnya sedikit berbeda tergantung pada apakah sistem operasi komputer lokal Anda adalah Windows atau macOS X.

Untuk informasi selengkapnya tentang prasyarat untuk terhubung menggunakan RDP, lihatPrasyarat.

Windows

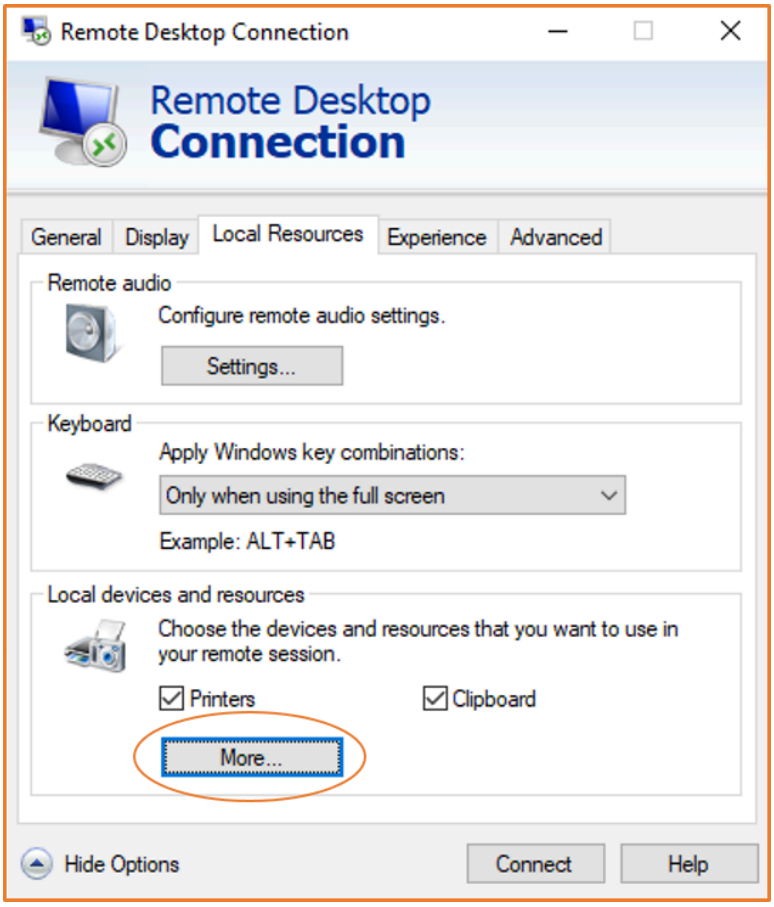
Untuk memetakan drive sesi jarak jauh ke folder lokal Anda di komputer Windows lokal Anda

1. Buka klien Remote Desktop Connection.
2. Pilih Tunjukkan Opsi.
3. Tambahkan nama host instance ke bidang Kom puter dan nama pengguna ke bidang Nama pengguna, sebagai berikut:

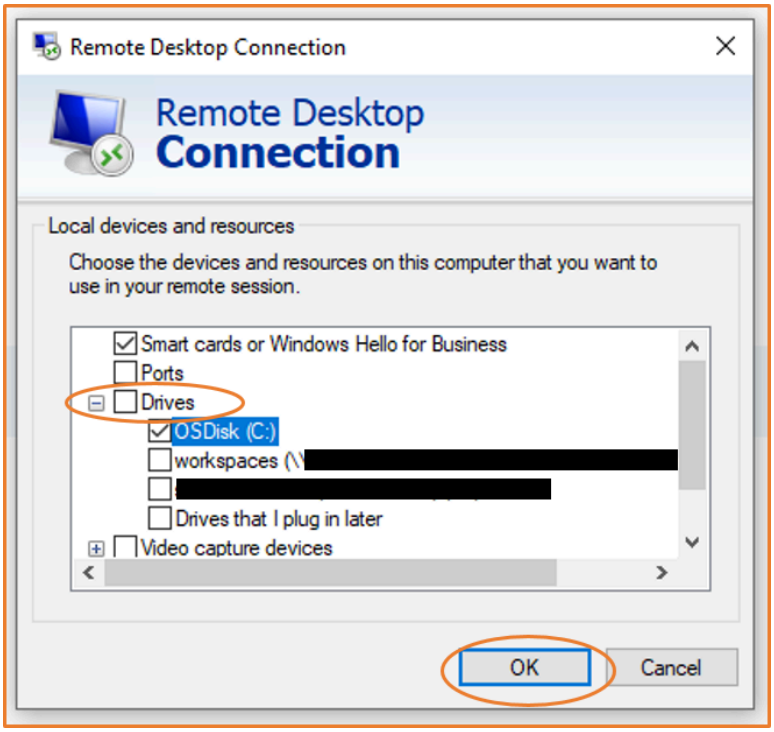
   a. Pada Pengaturan koneksi, pilih Buka... , lalu jelajahi file pintasan RDP yang Anda unduh dari konsol Amazon EC2. File tersebut berisi nama host DNS IPv4 Publik, yang mengidentifikasi instans, dan nama pengguna Administrator.

   b. Pilih file dan pilih Buka. Bidang Computer and User name diisi dengan nilai-nilai dari file shortcut RDP.

   c. Pilih Simpan.
4. Pilih tab Sumber Daya Lokal.
5. Di bawah Perangkat lokal dan sumber daya, pilih Selengkapnya...




6. Buka Drive dan pilih drive lokal untuk dipetakan ke instans Windows Anda.
7. Pilih OKE.

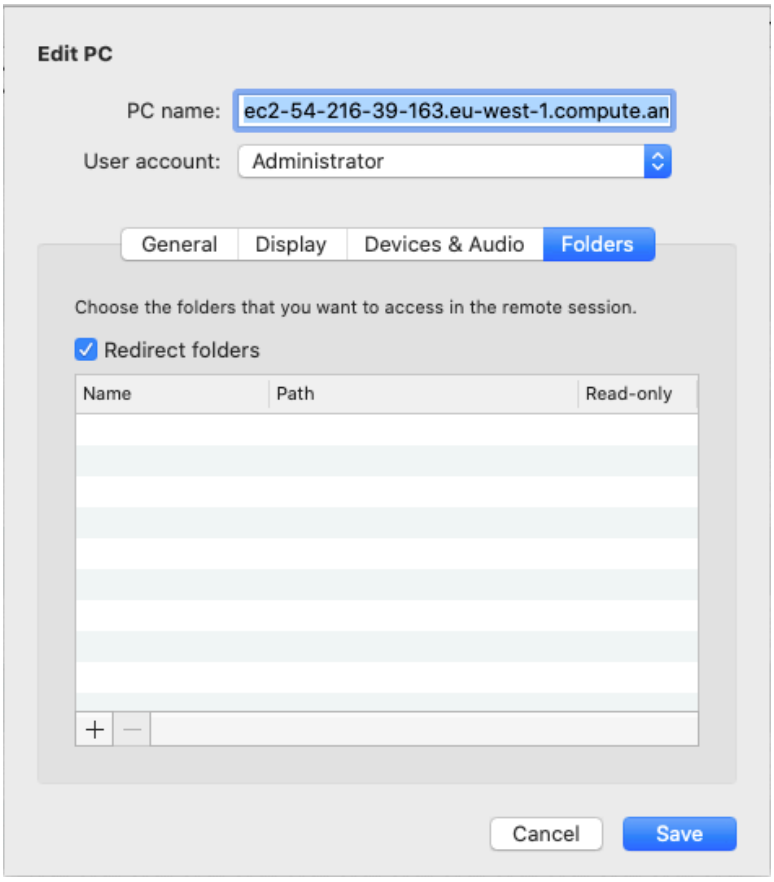



8. Pilih Hubungkan untuk terhubung ke instans Windows Anda.

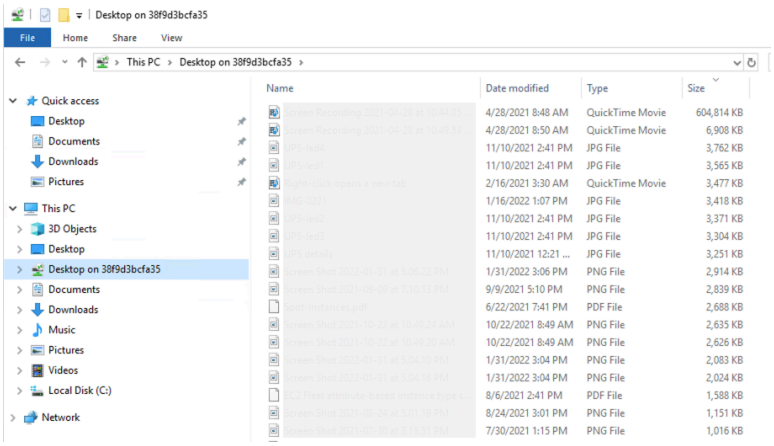
macOS X

Untuk memetakan drive sesi jarak jauh ke folder lokal Anda di komputer macOS X lokal Anda

1. Buka klien Remote Desktop Connection.
2. Jelajahi file RDP yang Anda unduh dari konsol Amazon EC2 (saat awal Anda terhubung ke instans), dan seret file ke klien Remote Desktop Connection.
3. Buka menu konteks (klik kanan) untuk file RDP, dan pilih Edit.
4. Pilih tab F older, dan pilih kotak centang Folder Arahkan.




5. Pilih ikon + di kiri bawah, jelajahi ke folder untuk memetakan, dan pilih Buka. Ulangi langkah ini untuk setiap folder untuk dipetakan.
6. Pilih Simpan.
7. Pilih Hubungkan untuk terhubung ke instans Windows Anda. Anda akan dimintai kata sandi.
8. Pada contoh, di File Explorer, perluas PC ini, dan temukan folder bersama tempat Anda dapat mengakses file lokal Anda. Pada tangkapan layar berikut, folder Desktop di komputer lokal dipetakan ke drive sesi jarak jauh pada instans.



Untuk informasi selengkapnya tentang membuat perangkat lokal tersedia untuk sesi jarak jauh di komputer Mac, lihat Memulai dengan klien macOS.

## Sambungkan ke instans Amazon EC2 menggunakan Pengelola Sesi

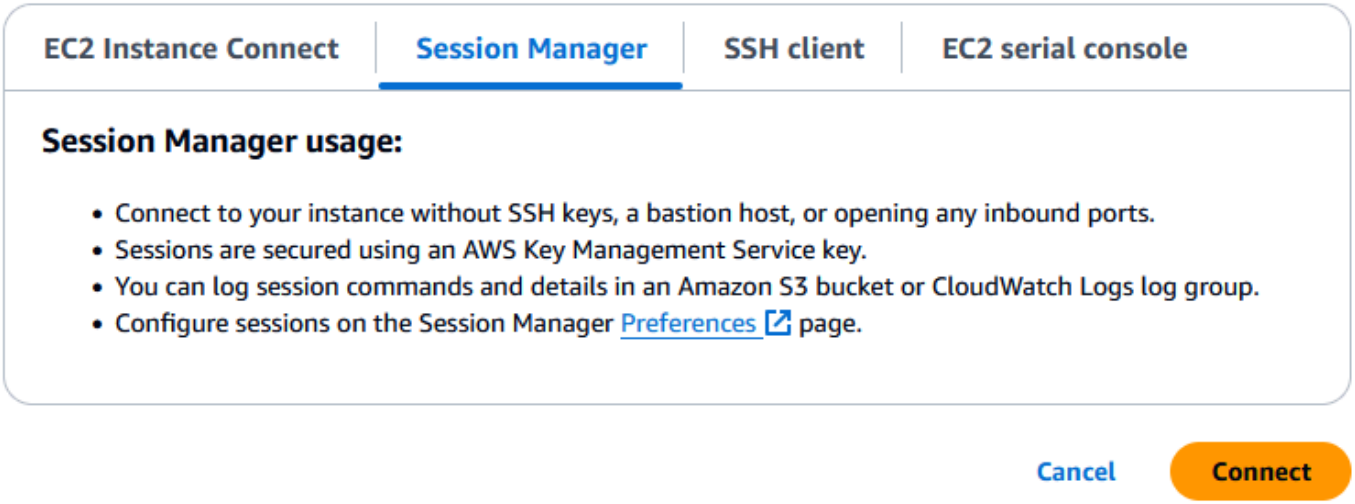
Session Manager adalah AWS Systems Manager kemampuan yang dikelola sepenuhnya untuk mengelola instans Amazon EC2 Anda melalui shell interaktif berbasis browser, atau melalui. AWS CLI Anda dapat menggunakan Session Manager untuk memulai sesi dengan sebuah instans di akun Anda. Setelah sesi dimulai, Anda dapat menjalankan perintah interaktif pada instance seperti yang Anda lakukan untuk jenis koneksi lainnya. Untuk informasi lebih lanjut tentang penggunaan Session Manager, lihat AWS Systems Manager Session Manager di Panduan Pengguna AWS Systems Manager .

Prasyarat

Sebelum mencoba menyambung ke instance menggunakan Pengelola Sesi, Anda harus menyelesaikan langkah-langkah penyiapan yang diperlukan. Misalnya, instance harus dikelola oleh SSM dan harus memiliki peran IAM yang terlampir dengan kebijakan. AmazonSSMManagedInstanceCore  Untuk informasi selengkapnya, lihat Menyiapkan Session Manager.

Untuk menyambung ke instans Amazon EC2 menggunakan Pengelola Sesi di konsol Amazon EC2

1. Buka konsol Amazon EC2 di  https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Hubungkan.
4. Untuk metode koneksi, pilih Pengel  ola Sesi.
5. Pilih  Connect  untuk memulai sesi.




Pemecahan masalah

Jika Anda menerima kesalahan bahwa Anda tidak berwenang untuk melakukan satu atau beberapa tindakan Manajer Sistem (`ssm:command-name`), Anda harus memperbarui kebijakan untuk memungkinkan Anda memulai sesi dari konsol Amazon EC2. Untuk informasi dan instruksi lebih lanjut, lihat Kebijakan IAM default Quickstart untuk Session Manager di Panduan Pengguna AWS Systems Manager .

## Hubungkan ke instans Linux Anda menggunakan alamat IP publik dan EC2 Instance Connect

Amazon EC2 Instance Connect menyediakan cara aman untuk terhubung ke instans Linux Anda melalui Secure Shell (SSH). Dengan EC2 Instance Connect, Anda menggunakan kebijakan dan prinsip AWS Identity and Access Management (IAM) untuk mengontrol akses SSH ke instans Anda, menghilangkan kebutuhan untuk berbagi dan mengelola kunci SSH. Semua permintaan koneksi menggunakan EC2 Instance Connect dicatat AWS CloudTrail sehingga Anda dapat mengaudit permintaan koneksi.

Anda dapat menggunakan EC2 Instance Connect untuk menyambung ke instans menggunakan konsol Amazon EC2 atau klien SSH pilihan Anda.

Saat Anda terhubung ke sebuah instans menggunakan EC2 Instance Connect, API EC2 Instance Connect mendorong kunci publik SSH ke metadata instans yang tetap bertahan selama 60 detik. Kebijakan IAM yang dilampirkan ke pengguna mengizinkan pengguna Anda untuk mendorong kunci publik ke metadata instans. Daemon SSH menggunakan `AuthorizedKeysCommand`

dan`AuthorizedKeysCommandUser`, yang dikonfigurasi saat EC2 Instance Connect diinstal, untuk mencari kunci publik dari metadata instans untuk otentikasi, dan menghubungkan Anda ke instance.

> **Tip**
>
> EC2 Instance Connect adalah salah satu opsi untuk terhubung ke instans Linux Anda. Untuk opsi lain, lihat Hubungkan ke instans Linux Anda, menggunakan SSH.. Untuk menyambung ke instance Windows, lihatHubungkan ke instans Windows Anda menggunakan RDP.

Harga

EC2 Instance Connect tersedia tanpa biaya tambahan.

Ketersediaan wilayah

EC2 Instance Connect tersedia di semua Wilayah AWS kecuali AWS European Sovereign Cloud (Germany). Hal ini tidak didukung di Zona Lokal.

Daftar Isi


- Tutorial: Selesaikan konfigurasi yang diperlukan untuk menyambung ke instans Anda menggunakan EC2 Instance Connect
- Prasyarat untuk EC2 Instance Connect
- Berikan izin IAM untuk EC2 instans Connect
- Instal EC2 instans Connect pada instans EC2 Anda
- Menghubungkan ke instans Linux menggunakan EC2 Instance Connect
- Hapus instalasi EC2 Instance Connect


Untuk posting blog yang membahas cara meningkatkan keamanan host bastion Anda menggunakan EC2 Instance Connect, lihat Mengamankan host bastion dengan Amazon EC2 Instance Connect.

## Tutorial: Selesaikan konfigurasi yang diperlukan untuk menyambung ke instans Anda menggunakan EC2 Instance Connect

Untuk menyambung ke instans menggunakan EC2 Instance Connect di konsol Amazon EC2, Anda harus terlebih dahulu menyelesaikan konfigurasi prasyarat yang memungkinkan Anda untuk berhasil menyambung ke instans. Tujuan dari tutorial ini adalah untuk memandu Anda melalui tugas untuk menyelesaikan konfigurasi prasyarat.

Ikhtisar tutorial

Dalam tutorial ini, Anda akan menyelesaikan empat tugas berikut:

- Tugas 1: Memberikan izin yang diperlukan untuk menggunakan EC2 Instance Connect

  Pertama, Anda akan membuat kebijakan IAM yang berisi izin IAM yang memungkinkan Anda untuk mendorong kunci publik ke metadata instance. Anda akan melampirkan kebijakan ini ke identitas IAM Anda (pengguna, grup pengguna, atau peran) sehingga identitas IAM Anda mendapatkan izin ini.

- Tugas 2: Izinkan lalu lintas masuk dari layanan EC2 Instance Connect ke instans Anda

  Kemudian Anda akan membuat grup keamanan yang memungkinkan lalu lintas dari layanan EC2 Instance Connect ke instans Anda. Ini diperlukan saat Anda menggunakan EC2 Instance Connect di konsol Amazon EC2 untuk menyambung ke instans Anda.

- Tugas 3: Luncurkan instance Anda

  Anda kemudian akan meluncurkan instance EC2 menggunakan AMI yang sudah diinstal sebelumnya dengan EC2 Instance Connect dan Anda akan menambahkan grup keamanan yang Anda buat pada langkah sebelumnya.

- Tugas 4: Hubungkan ke instance Anda

  Terakhir, Anda akan menggunakan EC2 Instance Connect di konsol Amazon EC2 untuk terhubung ke instans Anda. Jika Anda dapat terhubung, maka Anda dapat yakin bahwa konfigurasi prasyarat yang Anda selesaikan di Tugas 1, 2, dan 3 berhasil.

Tugas 1: Memberikan izin yang diperlukan untuk menggunakan EC2 Instance Connect

Saat Anda terhubung ke sebuah instans menggunakan EC2 Instance Connect, API EC2 Instance Connect mendorong kunci publik SSH ke metadata instans yang tetap bertahan selama 60 detik. Anda memerlukan kebijakan IAM yang dilampirkan ke identitas IAM Anda (pengguna, grup pengguna, atau peran) untuk memberi Anda izin yang diperlukan untuk mendorong kunci publik ke metadata instans.

Tujuan tugas

Anda akan membuat kebijakan IAM yang memberikan izin untuk mendorong kunci publik ke instance. Tindakan spesifik yang diizinkan adalah`ec2-instance-connect:SendSSHPublicKey`. Anda juga

harus mengizinkan `ec2:DescribeInstances` tindakan tersebut sehingga Anda dapat melihat dan memilih instans Anda di konsol Amazon EC2.

Setelah membuat kebijakan, Anda akan melampirkan kebijakan ke identitas IAM Anda (pengguna, grup pengguna, atau peran) sehingga identitas IAM Anda mendapatkan izin.

Anda akan membuat kebijakan yang dikonfigurasi sebagai berikut:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeInstances",
            "Resource": "*"
        }
    ]
}
```

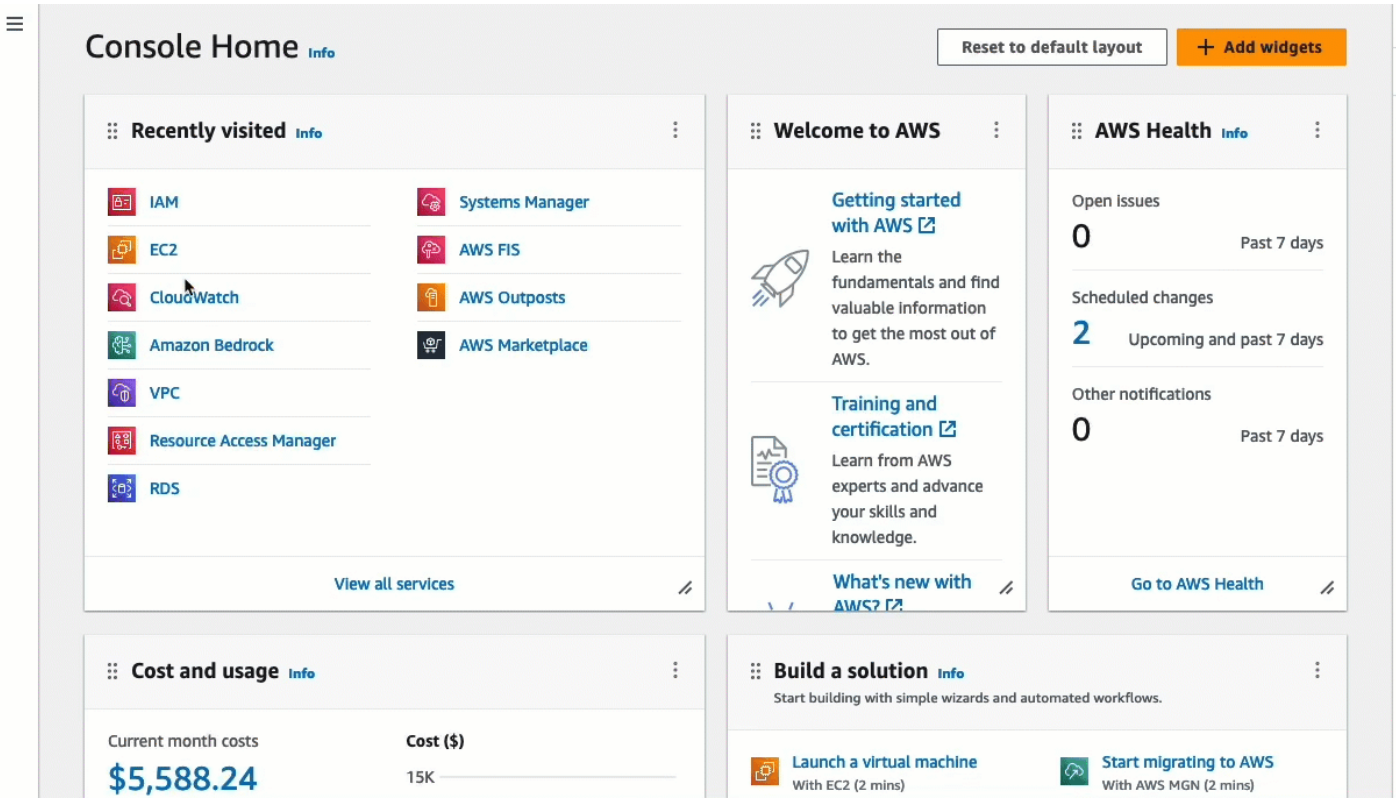
**Important**

Kebijakan IAM yang dibuat dalam tutorial ini adalah kebijakan yang sangat permisif; ini memungkinkan Anda untuk terhubung ke instance apa pun menggunakan nama pengguna AMI apa pun. Tutorial ini menggunakan kebijakan yang sangat permisif ini untuk menjaga tutorial tetap sederhana dan fokus pada konfigurasi spesifik yang diajarkan tutorial ini. Namun, di lingkungan produksi, sebaiknya kebijakan IAM dikonfigurasi untuk memberikan izin hak istimewa ter kecil. Untuk contoh kebijakan IAM, lihat Berikan izin IAM untuk EC2 instans Connect.

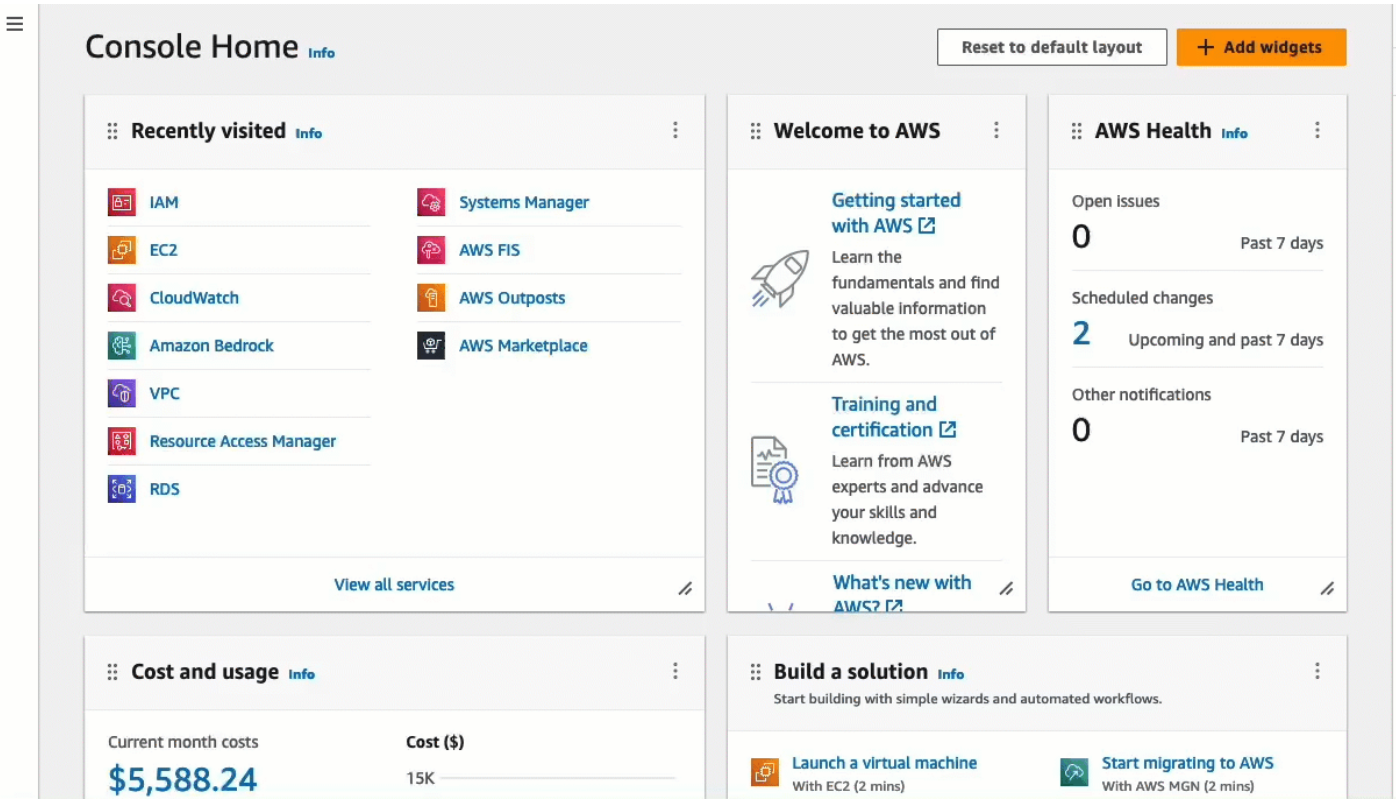
Untuk membuat dan melampirkan kebijakan IAM yang memungkinkan Anda menggunakan EC2 Instance Connect untuk menyambung ke instans Anda

1. Pertama buat kebijakan IAM
   a. Buka konsol IAM di https://console.aws.amazon.com/iam/.
   b. Di panel navigasi, pilih Kebijakan.
   c. Pilih Buat kebijakan.
   d. Pada halaman Tentukan izin, lakukan hal berikut:
      i. Untuk Layanan, pilih EC2 Instance Connect.
      ii. Di bawah Tindakan diizinkan, di bidang pencarian mulai mengetik **send** untuk menampilkan tindakan yang relevan, lalu pilih SendSSHPublicKey.
      iii. Di bawah Sumber Daya, pilih Semua. Untuk lingkungan produksi, kami sarankan menentukan instance dengan ARN-nya, tetapi untuk tutorial ini, Anda mengizinkan semua instance.
      iv. Pilih Tambahkan izin lainnya.
      v. Untuk Layanan , pilih EC2.
      vi. Di bawah Tindakan diizinkan, di bidang pencarian mulai mengetik **describein** untuk menampilkan tindakan yang relevan, lalu pilih DescribeInstances.
      vii. Pilih Berikutnya.
   e. Pada halaman Tinjau dan buat, lakukan hal berikut:
      i. Untuk Nama kebijakan, masukkan nama untuk kebijakan tersebut.
      ii. Pilih Buat kebijakan.
2. Kemudian lampirkan kebijakan ke identitas Anda
   a. Di konsol IAM, di panel navigasi, pilih Kebijakan.
   b. Dalam daftar kebijakan, pilih tombol opsi di sebelah nama kebijakan yang Anda buat. Anda dapat menggunakan kotak pencarian untuk memfilter daftar kebijakan.
   c. Pilih Tindakan, Lampirkan.
   d. Di bawah entitas IAM, pilih kotak centang di sebelah identitas Anda (pengguna, grup pengguna, atau peran). Anda dapat menggunakan kotak pencarian untuk memfilter daftar entitas.
   e. Pilih Lampirkan kebijakan.

## Melihat animasi: Membuat kebijakan IAM




## Melihat animasi: Melampirkan kebijakan IAM



### Tugas 2: Izinkan lalu lintas masuk dari layanan EC2 Instance Connect ke instans Anda

Saat Anda menggunakan Koneksi Instans EC2 di konsol Amazon EC2 untuk menyambung ke instans, lalu lintas yang harus diizinkan untuk mencapai instans adalah lalu lintas dari layanan EC2 Instance Connect. Ini berbeda dengan menghubungkan dari komputer lokal Anda ke instance; dalam hal ini, Anda harus mengizinkan lalu lintas dari komputer lokal Anda ke instance Anda. Untuk mengizinkan lalu lintas dari layanan EC2 Instance Connect, Anda harus membuat grup keamanan yang mengizinkan lalu lintas SSH masuk dari rentang alamat IP untuk layanan EC2 Instance Connect.

AWS menggunakan daftar awalan untuk mengelola rentang alamat IP. Nama-nama daftar awalan EC2 Instance Connect adalah sebagai berikut, dengan *region* diganti dengan kode Wilayah:

- Nama daftar awalan IPv4: `com.amazonaws.region.ec2-instance-connect`
- Nama daftar awalan IPv6: `com.amazonaws.region.ipv6.ec2-instance-connect`

#### Tujuan tugas

Anda akan membuat grup keamanan yang memungkinkan lalu lintas SSH masuk pada port 22 dari daftar awalan IPv4 di Wilayah tempat instance Anda berada.

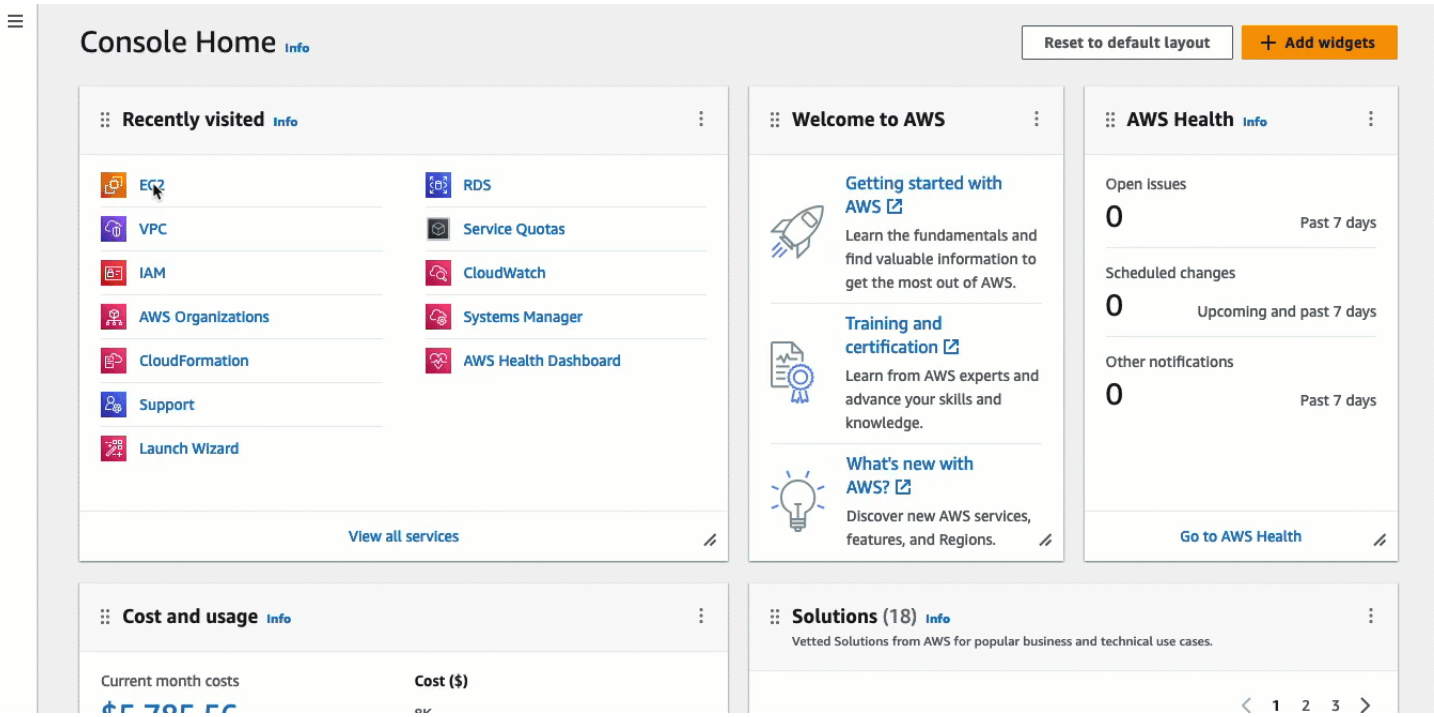
#### Untuk membuat grup keamanan yang mengizinkan lalu lintas masuk dari layanan EC2 Instance Connect ke instans Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Security Groups (Grup Keamanan).
3. Pilih Buat grup keamanan.
4. Di Detail dasar, lakukan hal berikut:
   a. Untuk Nama grup keamanan, masukkan nama yang berarti untuk grup keamanan Anda.
   b. Untuk Deskripsi, masukkan deskripsi yang berarti untuk grup keamanan Anda.
5. Di bawah aturan masuk, lakukan hal berikut:
   a. Pilih Tambahkan aturan.
   b. Untuk Tipe, pilih SSH.
   c. Untuk Sumber, tinggalkan Custom.
   d. Di bidang di sebelah Sumber, pilih daftar awalan untuk EC2 Instance Connect.

Misalnya, jika instance Anda berada di Wilayah AS Timur (Virginia Utara) (`us-east-1`) dan pengguna Anda akan terhubung ke alamat IPv4 publiknya, pilih daftar awalan berikut: com.amazonaws.us-east-1.ec2-instance-connect

6. Pilih Buat grup keamanan.

Melihat animasi: Buat grup keamanan




Tugas 3: Luncurkan instance Anda

Saat meluncurkan instance, Anda harus menentukan AMI yang berisi informasi yang diperlukan untuk meluncurkan instance. Anda dapat memilih untuk meluncurkan instans dengan atau tanpa EC2 Instance Connect yang sudah diinstal sebelumnya. Dalam tugas ini, Anda menentukan AMI yang sudah diinstal sebelumnya dengan EC2 Instance Connect.

Jika Anda meluncurkan instans tanpa EC2 Instance Connect yang sudah diinstal sebelumnya, dan Anda ingin menggunakan EC2 Instance Connect untuk menyambung ke instans, Anda harus melakukan langkah-langkah konfigurasi tambahan. Langkah-langkah ini berada di luar lingkup tutorial ini.

Tujuan tugas

Anda akan meluncurkan instance dengan Amazon Linux 2023 AMI, yang sudah diinstal sebelumnya dengan EC2 Instance Connect. Anda juga harus menentukan grup keamanan yang Anda buat sebelumnya sehingga Anda dapat menggunakan EC2 Instance Connect di konsol Amazon EC2 untuk menyambung ke instans Anda. Karena Anda akan menggunakan EC2 Instance Connect untuk menyambung ke instans, yang mendorong kunci publik ke metadata instans Anda, Anda tidak perlu menentukan kunci SSH saat meluncurkan instans.

Untuk meluncurkan instans yang dapat menggunakan EC2 Instance Connect di konsol Amazon EC2 untuk koneksi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di bilah navigasi di bagian atas layar, Wil AWS ayah saat ini ditampilkan (misalnya, Irlandia). Pilih Wilayah tempat peluncuran instans Anda. Pilihan ini penting karena Anda membuat grup keamanan yang mengizinkan lalu lintas untuk Wilayah tertentu, jadi Anda harus memilih Wilayah yang sama untuk meluncurkan instans Anda.
3. Dari dasbor konsol Amazon EC2, pilih Luncurkan instans.
4. (Opsional) Pada Nama dan tanda, untuk Nama, masukkan nama deskriptif untuk instans Anda.
5. Di bawah Aplikasi dan Gambar OS (Gambar Mesin Amazon), pilih Mulai Cepat. Amazon Linux dipilih secara default. Di bawah Amazon Machine Image (AMI), Amazon Linux 2023 AMI dipilih secara default. Pertahankan pilihan default untuk tugas ini.
6. Di bawah Jenis Inst ance, untuk Jenis Inst ance, pertahankan pilihan default, atau pilih jenis instans yang berbeda.
7. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih L anjutkan tanpa pasangan kunci (Tidak disarankan). Saat Anda menggunakan EC2 Instance Connect untuk menyambung ke instans, EC2 Instance Connect mendorong pasangan kunci ke metadata instans, dan pasangan kunci inilah yang digunakan untuk koneksi.
8. Pada Pengaturan jaringan, lakukan hal berikut:

   a. Untuk IP Auto-assign publik, tinggalkan Aktifkan.

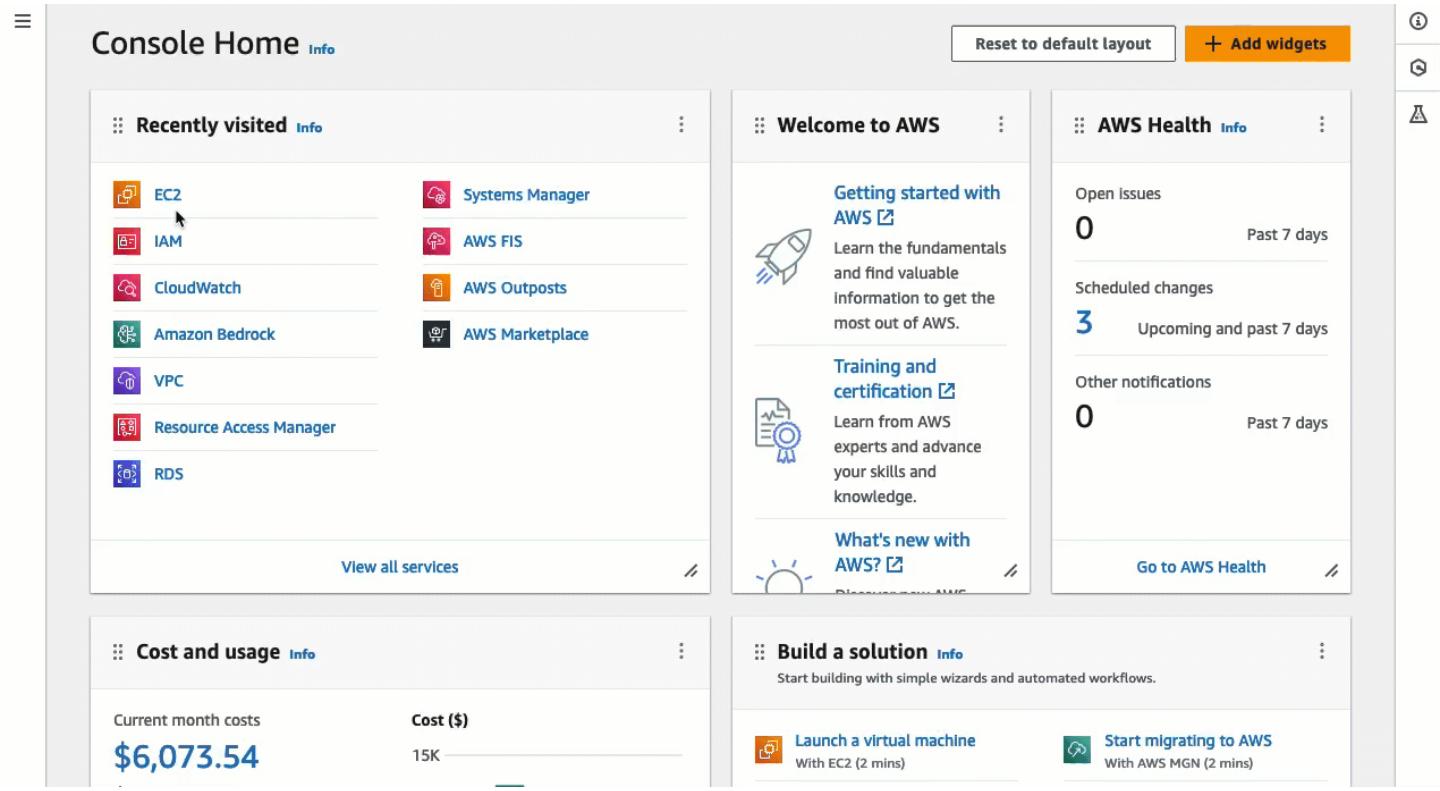
   > **Note**
   >
   > Untuk menggunakan EC2 Instance Connect di konsol Amazon EC2 untuk menyambung ke instans, instans harus memiliki alamat IPv4 atau IPv6 publik.

   b. Untuk Firewall (grup keamanan), pilih Pilih grup keamanan yang ada.

c. Di bawah Grup keamanan umum, pilih grup keamanan yang Anda buat sebelumnya.

9. Di panel Ringkasan, pilih Luncurkan instans.

Melihat animasi: Luncurkan instance Anda




Tugas 4: Hubungkan ke instance Anda

Saat Anda terhubung ke sebuah instans menggunakan EC2 Instance Connect, API EC2 Instance Connect mendorong kunci publik SSH ke metadata instans yang tetap bertahan selama 60 detik. Daemon SSH menggunakan `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser` untuk mencari kunci publik dari metadata instance untuk otentikasi, dan menghubungkan Anda ke instance.

Tujuan tugas

Dalam tugas ini, Anda akan terhubung ke instans menggunakan EC2 Instance Connect di konsol Amazon EC2. Jika Anda menyelesaikan prasyarat Tugas 1, 2, dan 3, koneksi harus berhasil.

Langkah-langkah untuk menyambung ke instance Anda

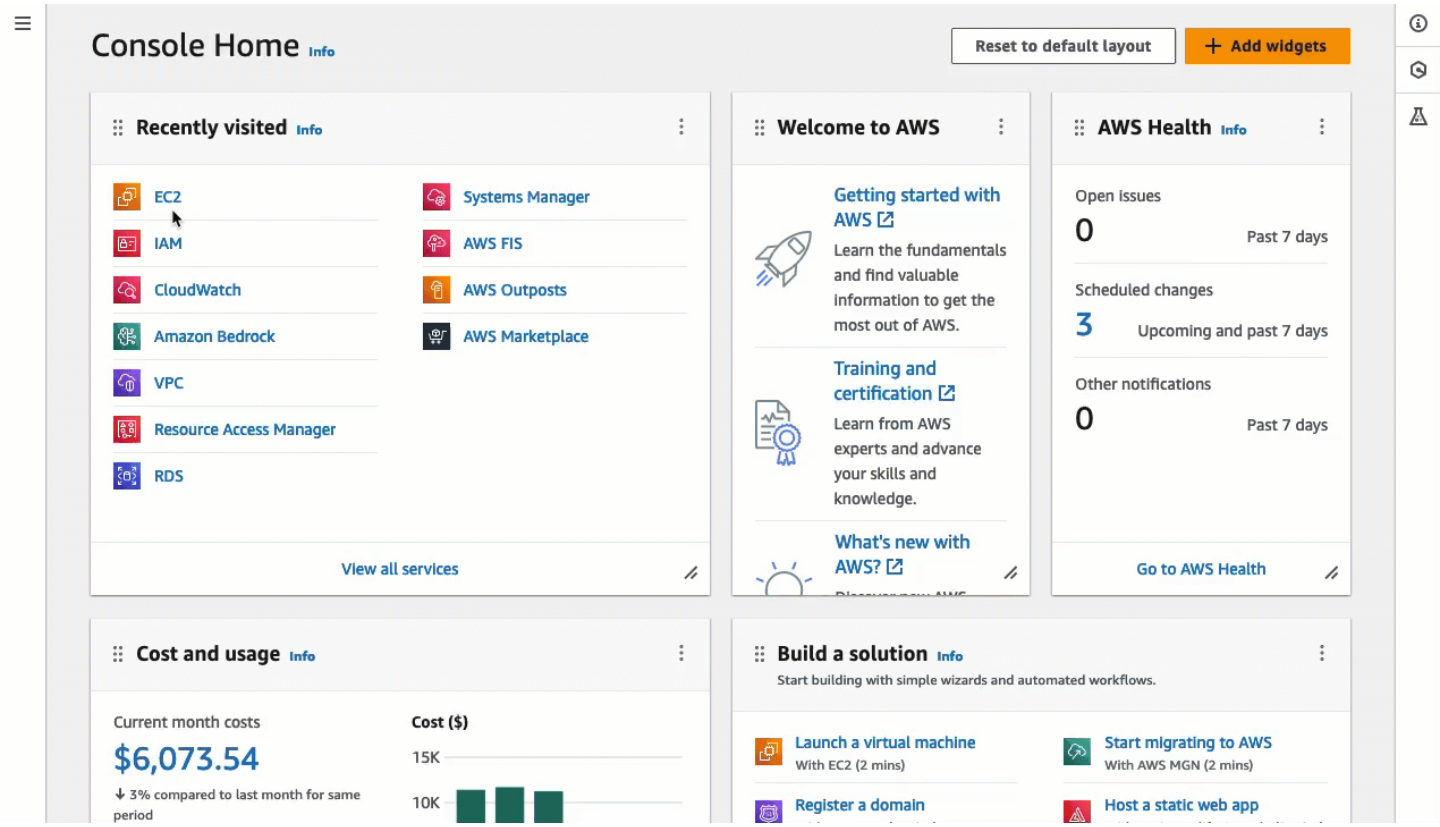
Gunakan langkah-langkah berikut untuk menyambung ke instance Anda. Untuk melihat animasi langkah-langkahnya, lihatMelihat animasi: Sambungkan ke instance Anda.

Untuk menghubungkan instans menggunakan EC2 Instance Connect di konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di bilah navigasi di bagian atas layar, Wil AWS ayah saat ini ditampilkan (misalnya, Irlandia). Pilih Wilayah tempat instance Anda berada.
3. Di panel navigasi, pilih Instans.
4. Pilih instance Anda dan pilih Hubun gkan.
5. Pilih tab EC2 Instance Connect.
6. Pilih Hubungkan menggunakan IP Publik.
7. Pilih Hubungkan.

   Jendela terminal terbuka di browser, dan Anda terhubung ke instance Anda.

Melihat animasi: Sambungkan ke instance Anda




## Prasyarat untuk EC2 Instance Connect

Berikut ini adalah prasyarat untuk menginstal dan menggunakan EC2 Instance Connect:

- Menginstal EC2 Instance Connect
- Pastikan konektivitas jaringan
- Izinkan lalu lintas keluar ke proxy EC2 Instance Connect
- Izinkan lalu lintas SSH masuk
- Berikan izin
- Instal klien SSH di komputer lokal Anda
- Memenuhi persyaratan nama pengguna

### Menginstal EC2 Instance Connect

Untuk menggunakan EC2 Instance Connect, instans harus menginstal EC2 Instance Connect. Anda dapat meluncurkan instans menggunakan AMI yang sudah diinstal sebelumnya dengan EC2 Instance Connect, atau Anda dapat menginstal EC2 Instance Connect pada instans yang diluncurkan dengan AMI yang didukung. Untuk informasi selengkapnya, lihat Instal EC2 instans Connect pada instans EC2 Anda.

### Pastikan konektivitas jaringan

Instans dapat dikonfigurasi untuk memungkinkan pengguna terhubung ke instans Anda melalui internet atau melalui alamat IP pribadi instans. Tergantung pada bagaimana pengguna Anda akan terhubung ke instans Anda menggunakan EC2 Instance Connect, Anda harus mengonfigurasi akses jaringan berikut:

- Jika pengguna Anda akan terhubung ke instans Anda melalui internet, maka instance Anda harus memiliki alamat IPv4 atau IPv6 publik dan berada di subnet publik dengan rute ke internet. Jika Anda belum memodifikasi subnet publik default Anda, maka itu berisi rute ke internet hanya untuk IPv4, dan bukan untuk IPv6. Untuk informasi selengkapnya, lihat Meng aktifkan akses internet VPC menggunakan gateway internet di Panduan Pengguna Amazon VPC.
- Jika pengguna Anda akan terhubung ke instans Anda melalui alamat IPv4 pribadi instans, maka Anda harus membuat konektivitas jaringan pribadi ke VPC Anda, seperti menggunakan AWS Direct Connect, AWS Site-to-Site VPN, atau peering VPC, sehingga pengguna Anda dapat mencapai alamat IP pribadi instans.

Jika instans Anda tidak memiliki alamat IPv4 atau IPv6 publik dan Anda memilih untuk tidak mengonfigurasi akses jaringan seperti yang dijelaskan di atas, Anda dapat mempertimbangkan EC2

Instance Connect Endpoint sebagai alternatif untuk EC2 Instance Connect. Dengan EC2 Instance Connect Endpoint, Anda dapat terhubung ke instance menggunakan SSH atau RDP meskipun instans tidak memiliki alamat IPv4 atau IPv6 publik. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda menggunakan konsol Amazon EC2.

Izinkan lalu lintas keluar ke proxy EC2 Instance Connect

Saat Anda menyambung ke instans menggunakan konsol Amazon EC2, browser Anda membuat WebSocket koneksi pada port 443 ke layanan proxy EC2 Instance Connect di titik akhir. Region-specific

Format titik akhir tergantung pada partisi:

- Wilayah AWS Komersional dan Wilayah AWS GovCloud (AS) menggunakan format titik akhir:`prod.region.oneclickv2-proxy.ec2.aws.dev`.
- Wilayah China menggunakan format titik akhir:`prod.region.oneclickv2-proxy.ec2.a2z.org.cn`.

Contoh berikut menunjukkan titik akhir untuk setiap partisi:

- Untuk Wilayah Timur AS (Virginia Utara), titik akhir adalah`prod.us-east-1.oneclickv2-proxy.ec2.aws.dev`.
- Untuk Wilayah China (Beijing), titik akhirnya adalah`prod.cn-north-1.oneclickv2-proxy.ec2.a2z.org.cn`.

> **Note**
>
> Browser Anda harus dapat mencapai titik akhir ini. Jika firewall atau VPN memblokir koneksi, koneksi gagal.

Izinkan lalu lintas SSH masuk

Saat menggunakan konsol Amazon EC2 untuk menyambung ke instans

Saat pengguna terhubung ke instans menggunakan konsol Amazon EC2, lalu lintas yang harus diizinkan untuk mencapai instans adalah lalu lintas dari layanan EC2 Instance Connect. Layanan diidentifikasi oleh rentang alamat IP tertentu, yang dikel AWS ola melalui daftar awalan. Anda harus

membuat grup keamanan yang memungkinkan lalu lintas SSH masuk dari layanan EC2 Instance Connect. Untuk mengonfigurasi ini, untuk aturan masuk, di bidang di sebelah Sumber, pilih daftar awalan EC2 Instance Connect.

AWS menyediakan daftar awalan terkelola yang berbeda untuk alamat IPv4 dan IPv6 untuk setiap Wilayah. Nama-nama daftar awalan EC2 Instance Connect adalah sebagai berikut, dengan *region* diganti dengan kode Wilayah:

- Nama daftar awalan IPv4: `com.amazonaws.region.ec2-instance-connect`
- Nama daftar awalan IPv6: `com.amazonaws.region.ipv6.ec2-instance-connect`

Untuk petunjuk untuk membuat grup keamanan, lihatTugas 2: Izinkan lalu lintas masuk dari layanan EC2 Instance Connect ke instans Anda. Untuk informasi selengkapnya, lihat Daftar AWS awalan yang dikelola tersedia di Panduan Pengguna Amazon VPC.

Saat menggunakan CLI atau SSH untuk terhubung ke instance

Pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas SSH masuk pada port 22 dari alamat IP Anda atau dari jaringan Anda. Grup keamanan default untuk VPC tidak mengizinkan lalu lintas SSH masuk secara default. Grup keamanan yang dibuat oleh wizard peluncuran instans memungkinkan lalu lintas SSH masuk secara default. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.

Berikan izin

Anda harus memberikan izin yang diperlukan untuk setiap pengguna IAM yang akan menggunakan EC2 Instance Connect untuk terhubung ke instans. Untuk informasi selengkapnya, lihat Berikan izin IAM untuk EC2 instans Connect.

Instal klien SSH di komputer lokal Anda

Jika pengguna Anda akan terhubung menggunakan SSH, mereka harus memastikan bahwa komputer lokal mereka memiliki klien SSH.

Komputer lokal milik pengguna kemungkinan besar telah menginstal klien SSH secara default. Mereka dapat memeriksa klien SSH dengan mengetik ssh di baris perintah. Jika komputer lokal mereka tidak mengenali perintah tersebut, mereka dapat menginstal klien SSH. Untuk informasi tentang menginstal klien SSH di Linux atau macOS X, lihat http://www.openssh.com. Untuk informasi tentang instalasi klien SSH di Windows 10, lihat OpenSSH di Windows.

Tidak perlu menginstal klien SSH di komputer lokal jika pengguna Anda hanya menggunakan konsol Amazon EC2 untuk terhubung ke instance.

Memenuhi persyaratan nama pengguna

Saat menggunakan EC2 Instance Connect untuk menyambung ke instans, nama pengguna harus memenuhi persyaratan berikut:

- Karakter pertama: Harus berupa huruf (`A-Z`,`a-z`), digit (`0-9`), atau garis bawah (`_`)
- Karakter selanjutnya: Dapat berupa huruf (`A-Z`, `a-z`), digit (`0-9`), atau karakter berikut: `@ . _ -`
- Panjang minimum: 1
- Panjang maksimal: 31 karakter

## Berikan izin IAM untuk EC2 instans Connect

Untuk terhubung ke instans menggunakan EC2 Instance Connect, Anda harus membuat kebijakan IAM yang memberikan izin kepada pengguna Anda untuk tindakan dan kondisi berikut:

- Tindakan `ec2-instance-connect:SendSSHPublicKey` – Memberikan izin untuk mendorong kunci publik ke sebuah instans.
- Kondisi `ec2:osuser` – Menentukan nama pengguna OS yang dapat mendorong kunci publik ke sebuah instans. Gunakan nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instance. Nama pengguna default untuk AL2023 dan Amazon Linux 2 adalah`ec2-user`, dan untuk Ubuntu adalah`ubuntu`.
- Tindakan `ec2:DescribeInstances` – Wajib saat menggunakan konsol EC2 karena wrapper memanggil tindakan ini. Pengguna mungkin sudah memiliki izin untuk memanggil tindakan ini dari kebijakan lain.
- `ec2:DescribeVpcs`tindakan - Diperlukan saat menghubungkan ke alamat IPv6.

Pertimbangkan untuk membatasi akses ke instans EC2 tertentu. Jika tidak, semua prinsipal IAM dengan izin untuk `ec2-instance-connect:SendSSHPublicKey` tindakan tersebut dapat terhubung ke semua instans EC2. Anda dapat membatasi akses dengan menentukan ARN sumber daya atau dengan menggunakan tanda sumber daya sebagai kunci syarat.

Untuk informasi selengkapnya, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2 instans Connect.

Untuk informasi tentang pembuatan kebijakan IAM, lihat Membuat kebijakan IAM di Panduan Pengguna IAM.

Izinkan pengguna untuk terhubung ke instans tertentu

Kebijakan IAM berikut memberikan izin untuk terhubung ke instans tertentu, yang diidentifikasi oleh ARN sumber daya mereka.

Dalam contoh kebijakan IAM berikut, tindakan dan kondisi berikut ditentukan:

- Tindakan `ec2-instance-connect:SendSSHPublicKey` ini memberi pengguna izin untuk terhubung ke dua instans, yang ditentukan oleh ARN sumber daya. Untuk memberikan izin kepada pengguna untuk terhubung ke semua instans EC2, ganti ARN sumber daya dengan wildcard `*`.
- `ec2:osuser`Kondisi memberikan izin untuk terhubung ke instance hanya jika *`ami-username`* ditentukan saat menghubungkan.
- Tindakan `ec2:DescribeInstances` ditentukan untuk memberikan izin kepada pengguna yang akan menggunakan konsol untuk terhubung ke instans Anda. Jika pengguna Anda hanya akan menggunakan klien SSH untuk terhubung ke instans Anda, maka Anda dapat menghapus `ec2:DescribeInstances`. Perhatikan bahwa tindakan API `ec2:Describe*` tidak mendukung izin tingkat sumber daya. Oleh karena itu, wildcard `*` dibutuhkan dalam `Resource`.
- T `ec2:DescribeVpcs` indakan ditentukan untuk memberikan izin kepada pengguna yang akan menggunakan konsol untuk terhubung ke instans Anda menggunakan alamat IPv6. Jika pengguna Anda hanya akan menggunakan alamat IPv4 publik, Anda dapat menghilangkannya. `ec2:DescribeVpcs` Perhatikan bahwa tindakan API `ec2:Describe*` tidak mendukung izin tingkat sumber daya. Oleh karena itu, wildcard `*` dibutuhkan dalam `Resource`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": [
                "arn:aws:ec2:us-
east-1:111122223333:instance/i-1234567890abcdef0",
                "arn:aws:ec2:us-east-1:111122223333:instance/i-0598c7d356eba48d7"
            ],
```

```
            "Condition": {
                "StringEquals": {
                    "ec2:osuser": "ami-username"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeVpcs"
            ],
            "Resource": "*"
        }
    ]
}
```

Izinkan pengguna untuk terhubung ke instans dengan tanda tertentu

Attribute-based kontrol akses (ABAC) adalah strategi otorisasi yang mendefinisikan izin berdasarkan tag yang dapat dilampirkan ke pengguna dan AWS sumber daya. Anda dapat menggunakan tanda sumber daya untuk mengontrol akses ke instans. Untuk informasi selengkapnya tentang menggunakan tag untuk mengontrol akses ke AWS sumber daya Anda, lihat Meng ontrol akses ke AWS sumber daya di Panduan Pengguna IAM.

Dalam contoh kebijakan IAM berikut, tindakan `ec2-instance-connect:SendSSHPublicKey` memberikan izin kepada pengguna untuk terhubung ke instans apa pun (ditunjukkan oleh wildcard `*` di ARN sumber daya) dengan syarat instans memiliki tanda sumber daya dengan kunci=`tag-key` dan value=`tag-value`.

Tindakan `ec2:DescribeInstances` ditentukan untuk memberikan izin kepada pengguna yang akan menggunakan konsol untuk terhubung ke instans Anda. Jika pengguna Anda hanya akan menggunakan klien SSH untuk terhubung ke instans Anda, Anda dapat menghilangkannya`ec2:DescribeInstances`. Perhatikan bahwa tindakan API `ec2:Describe*` tidak mendukung izin tingkat sumber daya. Oleh karena itu, wildcard `*` dibutuhkan dalam `Resource`.

T `ec2:DescribeVpcs` indakan ditentukan untuk memberikan izin kepada pengguna yang akan menggunakan konsol untuk terhubung ke instans Anda menggunakan alamat IPv6. Jika pengguna Anda hanya akan menggunakan alamat IPv4 publik, Anda dapat menghilangkannya.

`ec2:DescribeVpcs` Perhatikan bahwa tindakan API `ec2:Describe*` tidak mendukung izin tingkat sumber daya. Oleh karena itu, wildcard * dibutuhkan dalam `Resource`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/tag-key": "tag-value"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeVpcs"
            ],
            "Resource": "*"
        }
    ]
}
```

## Instal EC2 instans Connect pada instans EC2 Anda

Untuk menyambung ke instans Linux menggunakan EC2 Instance Connect, instans harus menginstal EC2 Instance Connect. Menginstal EC2 Instance Connect akan mengonfigurasi daemon SSH pada instans.

Untuk informasi selengkapnya tentang paket EC2 Instance Connect, lihat aws/aws -ec2-inst ance-connect-config di situs web. GitHub

> **Note**
>
> Jika Anda mengonfigurasi pengaturan `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser` untuk autentikasi SSH, instalasi EC2 Instance Connect tidak akan memperbaruinya. Akibatnya, Anda tidak dapat menggunakan EC2 Instance Connect.

### Instal prasyarat

Sebelum menginstal EC2 Instance Connect, pastikan Anda memenuhi prasyarat berikut.

- Verifikasi bahwa instance menggunakan salah satu dari berikut ini:
  - Amazon Linux 2 sebelum versi 2.0.20190618*
  - AL2023 AMI minimal atau Amazon ECS-optimized AMI
  - CentOS Stream 8 dan 9
  - macOS Sonoma sebelum 14.2.1, Ventura sebelum 13.6.3, dan Monterey sebelum 12.7.2 *
  - Red Hat Enterprise Linux (RHEL) 8 dan 9
  - Ubuntu 16.04 dan 18.04 *

  > **Tip**
  >
  > * Untuk Amazon Linux 2, macOS, dan Ubuntu: Jika Anda meluncurkan instans menggunakan versi yang lebih baru dari yang tercantum di atas, EC2 Instance Connect sudah diinstal sebelumnya dan tidak diperlukan instalasi manual.

- Verifikasi prasyarat umum untuk EC2 Instance Connect.

  Untuk informasi selengkapnya, lihat Prasyarat untuk EC2 Instance Connect.
- Verifikasi prasyarat untuk menghubungkan ke instans Anda menggunakan klien SSH di mesin lokal Anda.

  Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..
- Dapatkan ID instans.

Anda dapat memperoleh ID instans Anda menggunakan konsol Amazon EC2 (dari kolom ID instans). Jika Anda mau, Anda dapat menggunakan perintah describe-instance (AWS CLI) atau Get-EC2Instance (AWS Tools for Windows PowerShell).

### Instal EC2 Instance Connect secara manual

> **Note**
>
> Jika Anda meluncurkan instans menggunakan salah satu AMI berikut, EC2 Instance Connect sudah diinstal sebelumnya dan Anda dapat melewati prosedur ini:
>
> - AL2023 standar AMI
> - Amazon Linux 2 2.0.20190618 atau setelahnya
> - macOS Sonoma 14.2.1 atau yang lebih baru
> - macOS Ventura 13.6.3 atau yang lebih baru
> - macOS Monterey 12.7.2 atau yang lebih baru
> - Ubuntu 20.04 atau setelahnya

Gunakan salah satu prosedur berikut untuk menginstal EC2 Instance Connect, tergantung pada sistem operasi instans Anda.

#### Amazon Linux 2

Untuk menginstal EC2 Instance Connect pada instans yang diluncurkan dengan Amazon Linux 2

1. Hubungkan ke instans Anda menggunakan SSH.

   Ganti contoh nilai dalam perintah berikut dengan nilai Anda. Gunakan pasangan kunci SSH yang ditetapkan ke instance Anda saat Anda meluncurkannya dan nama pengguna default AMI yang Anda gunakan untuk meluncurkan instans Anda. Untuk Amazon Linux 2, nama pengguna defaultnya adalah`ec2-user`.

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat Hubungkan ke instans Linux Anda menggunakan klien SSH.

2. Instal paket EC2 instans Connect di instans Anda.

```
[ec2-user ~]$ sudo yum install ec2-instance-connect
```

Anda akan melihat tiga skrip baru di folder `/opt/aws/bin/`:

```
eic_curl_authorized_keys
eic_parse_authorized_keys
eic_run_authorized_keys
```

3. (Opsional) Memverifikasi bahwa EC2 Instance Connect berhasil diinstal pada instans Anda.

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config
```

EC2 Instance Connect berhasil diinstal jika baris `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser` berisi nilai-nilai berikut:

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` mengatur skrip `eic_run_authorized_keys` untuk mencari kunci dari metadata instans
- `AuthorizedKeysCommandUser` menetapkan pengguna sistem sebagai `ec2-instance-connect`

> **Note**
>
> Jika sebelumnya Anda mengonfigurasi `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser`, instalasi EC2 Instance Connect tidak akan mengubah nilainya dan Anda tidak akan dapat menggunakan EC2 Instance Connect.

### CentOS

Untuk menginstal EC2 Instance Connect pada instans yang diluncurkan dengan CentOS

1. Hubungkan ke instans Anda menggunakan SSH.

   Ganti contoh nilai dalam perintah berikut dengan nilai Anda. Gunakan pasangan kunci SSH yang ditetapkan ke instance Anda saat Anda meluncurkannya dan nama pengguna default AMI yang Anda gunakan untuk meluncurkan instans Anda. Untuk CentOS, nama pengguna default adalah `centos` atau`ec2-user`.

   ```
   $ ssh -i my_ec2_private_key.pem centos@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   Untuk informasi selengkapnya tentang terhubung ke instans Anda, lihat Hubungkan ke instans Linux Anda menggunakan klien SSH.

2. Jika Anda menggunakan proksi HTTP atau HTTPS, Anda harus mengatur variabel lingkungan `http_proxy` atau `https_proxy` di sesi shell saat ini.

   Jika Anda tidak menggunakan proksi, Anda dapat melewati langkah ini.

   - Untuk server proksi HTTP, jalankan perintah berikut:

     ```
     $ export http_proxy=http://hostname:port
     $ export https_proxy=http://hostname:port
     ```

   - Untuk server proksi HTTPS, jalankan perintah berikut:

     ```
     $ export http_proxy=https://hostname:port
     $ export https_proxy=https://hostname:port
     ```

3. Instal paket EC2 Instance Connect di instans Anda dengan menjalankan perintah berikut.

   File konfigurasi EC2 Instance Connect untuk CentOS disediakan dalam paket Red Hat Package Manager (RPM), dengan paket RPM yang berbeda untuk CentOS 8 dan CentOS 9 dan misalnya tipe yang berjalan pada Intel/AMD (x86_64) atau ARM (AArch64).

   Gunakan blok perintah untuk sistem operasi dan arsitektur CPU Anda.

   - CentOS 8

Intel/AMD (x86_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-
connect-2.0.0-5.rhel8.x86_64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
west-2.s3.us-west-2.amazonaws.com/latest/linux_arm64/ec2-instance-
connect-2.0.0-5.rhel8.aarch64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

- CentOS 9

Intel/AMD (x86_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-
connect-2.0.0-5.rhel9.x86_64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
```

```
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
west-2.s3.us-west-2.amazonaws.com/latest/linux_arm64/ec2-instance-
connect-2.0.0-5.rhel9.aarch64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

Anda akan melihat skrip baru berikut di folder `/opt/aws/bin/`:

```
eic_run_authorized_keys
```

4. (Opsional) Memverifikasi bahwa EC2 Instance Connect berhasil diinstal pada instans Anda.

   - Untuk CentOS 8:

     ```
     [ec2-user ~]$ sudo less /lib/systemd/system/sshd.service.d/ec2-instance-
     connect.conf
     ```

   - Untuk CentOS 9:

     ```
     [ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
     ```

   EC2 Instance Connect berhasil diinstal jika baris `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser` berisi nilai-nilai berikut:

   ```
   AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
   AuthorizedKeysCommandUser ec2-instance-connect
   ```

- `AuthorizedKeysCommand` mengatur skrip `eic_run_authorized_keys` untuk mencari kunci dari metadata instans
- `AuthorizedKeysCommandUser` menetapkan pengguna sistem sebagai `ec2-instance-connect`

> **Note**
>
> Jika sebelumnya Anda mengonfigurasi `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser`, instalasi EC2 Instance Connect tidak akan mengubah nilainya dan Anda tidak akan dapat menggunakan EC2 Instance Connect.

macOS

Untuk menginstal EC2 Instance Connect pada instans yang diluncurkan dengan macOS

1. Hubungkan ke instans Anda menggunakan SSH.

   Ganti contoh nilai dalam perintah berikut dengan nilai Anda. Gunakan pasangan kunci SSH yang ditetapkan ke instance Anda saat Anda meluncurkannya dan nama pengguna default AMI yang Anda gunakan untuk meluncurkan instans Anda. Untuk instance macOS, nama pengguna defaultnya adalah`ec2-user`.

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   Untuk informasi selengkapnya tentang terhubung ke instans Anda, lihat Hubungkan ke instans Linux Anda menggunakan klien SSH.

2. Perbarui Homebrew menggunakan perintah berikut. Pembaruan akan mencantumkan perangkat lunak yang diketahui Homebrew. Paket EC2 Instance Connect disediakan melalui Homebrew pada instans macOS.  Untuk informasi selengkapnya, lihat Memperbarui sistem operasi dan perangkat lunak pada instans Amazon EC2 Mac.

   ```
   [ec2-user ~]$ brew update
   ```

3. Instal paket EC2 instans Connect di instans Anda. Ini akan menginstal perangkat lunak dan mengonfigurasi sshd untuk menggunakannya.

```
[ec2-user ~]$ brew install ec2-instance-connect
```

Anda akan melihat skrip baru berikut di folder `/opt/aws/bin/`:

```
eic_run_authorized_keys
```

4. (Opsional) Memverifikasi bahwa EC2 Instance Connect berhasil diinstal pada instans Anda.

```
[ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
```

EC2 Instance Connect berhasil diinstal jika baris `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser` berisi nilai-nilai berikut:

```
AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` mengatur skrip `eic_run_authorized_keys` untuk mencari kunci dari metadata instans
- `AuthorizedKeysCommandUser` menetapkan pengguna sistem sebagai `ec2-instance-connect`

> **Note**
>
> Jika sebelumnya Anda mengonfigurasi `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser`, instalasi EC2 Instance Connect tidak akan mengubah nilainya dan Anda tidak akan dapat menggunakan EC2 Instance Connect.

RHEL

Untuk menginstal EC2 Instance Connect pada instans yang diluncurkan dengan Red Hat Enterprise Linux (RHEL)

1. Hubungkan ke instans Anda menggunakan SSH.

   Ganti contoh nilai dalam perintah berikut dengan nilai Anda. Gunakan pasangan kunci SSH yang ditetapkan ke instance Anda saat Anda meluncurkannya dan nama pengguna default

AMI yang Anda gunakan untuk meluncurkan instans Anda. Untuk RHEL, nama pengguna default adalah `ec2-user` atau `root`.

```
$ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
west-2.compute.amazonaws.com
```

Untuk informasi selengkapnya tentang terhubung ke instans Anda, lihat Hubungkan ke instans Linux Anda menggunakan klien SSH.

2. Jika Anda menggunakan proksi HTTP atau HTTPS, Anda harus mengatur variabel lingkungan `http_proxy` atau `https_proxy` di sesi shell saat ini.

   Jika Anda tidak menggunakan proksi, Anda dapat melewati langkah ini.

   - Untuk server proksi HTTP, jalankan perintah berikut:

     ```
     $ export http_proxy=http://hostname:port
     $ export https_proxy=http://hostname:port
     ```

   - Untuk server proksi HTTPS, jalankan perintah berikut:

     ```
     $ export http_proxy=https://hostname:port
     $ export https_proxy=https://hostname:port
     ```

3. Instal paket EC2 Instance Connect di instans Anda dengan menjalankan perintah berikut.

   File konfigurasi EC2 Instance Connect untuk RHEL disediakan dalam paket Red Hat Package Manager (RPM), dengan paket RPM yang berbeda untuk RHEL 8 dan RHEL 9 dan misalnya tipe yang berjalan pada Intel/AMD (x86_64) atau ARM (AArch64).

   Gunakan blok perintah untuk sistem operasi dan arsitektur CPU Anda.

   - RHEL 8

     Intel/AMD (x86_64)

     ```
     [ec2-user ~]$ mkdir /tmp/ec2-instance-connect
     [ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
     west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-
     connect-2.0.0-5.rhel8.x86_64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
     connect.rpm
     ```

```
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
west-2.s3.us-west-2.amazonaws.com/latest/linux_arm64/ec2-instance-
connect-2.0.0-5.rhel8.aarch64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

- RHEL 9

  Intel/AMD (x86_64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
west-2.s3.us-west-2.amazonaws.com/latest/linux_amd64/ec2-instance-
connect-2.0.0-5.rhel9.x86_64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_amd64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

  ARM (AArch64)

```
[ec2-user ~]$ mkdir /tmp/ec2-instance-connect
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-
west-2.s3.us-west-2.amazonaws.com/latest/linux_arm64/ec2-instance-
```

```
connect-2.0.0-5.rhel9.aarch64.rpm -o /tmp/ec2-instance-connect/ec2-instance-
connect.rpm
[ec2-user ~]$ curl https://amazon-ec2-instance-connect-us-west-2.s3.us-
west-2.amazonaws.com/latest/linux_arm64/ec2-instance-connect-
selinux-2.0.0-5.noarch.rpm -o /tmp/ec2-instance-connect/ec2-instance-connect-
selinux.rpm
[ec2-user ~]$ sudo yum install -y /tmp/ec2-instance-connect/ec2-instance-
connect.rpm /tmp/ec2-instance-connect/ec2-instance-connect-selinux.rpm
```

Anda akan melihat skrip baru berikut di folder `/opt/aws/bin/`:

```
eic_run_authorized_keys
```

4. (Opsional) Memverifikasi bahwa EC2 Instance Connect berhasil diinstal pada instans Anda.

   - Untuk RHEL 8:

     ```
     [ec2-user ~]$ sudo less /lib/systemd/system/sshd.service.d/ec2-instance-
     connect.conf
     ```

   - Untuk RHEL 9:

     ```
     [ec2-user ~]$ sudo less /etc/ssh/sshd_config.d/60-ec2-instance-connect.conf
     ```

   EC2 Instance Connect berhasil diinstal jika baris `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser` berisi nilai-nilai berikut:

   ```
   AuthorizedKeysCommand /opt/aws/bin/eic_run_authorized_keys %u %f
   AuthorizedKeysCommandUser ec2-instance-connect
   ```

   - `AuthorizedKeysCommand` mengatur skrip `eic_run_authorized_keys` untuk mencari kunci dari metadata instans
   - `AuthorizedKeysCommandUser` menetapkan pengguna sistem sebagai `ec2-instance-connect`

> **Note**
>
> Jika sebelumnya Anda mengonfigurasi `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser`, instalasi EC2 Instance Connect tidak akan mengubah nilainya dan Anda tidak akan dapat menggunakan EC2 Instance Connect.

Ubuntu

Untuk menginstal EC2 Instance Connect pada instans yang diluncurkan dengan Ubuntu 16.04 atau setelahnya

1. Hubungkan ke instans Anda menggunakan SSH.

   Ganti contoh nilai dalam perintah berikut dengan nilai Anda. Gunakan pasangan kunci SSH yang ditetapkan ke instance Anda saat Anda meluncurkannya dan gunakan nama pengguna default AMI yang Anda gunakan untuk meluncurkan instans Anda. Untuk AMI Ubuntu, nama penggunanya adalah`ubuntu`.

   ```
   $ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat Hubungkan ke instans Linux Anda menggunakan klien SSH.

2. (Opsional) Pastikan instans Anda memiliki AMI Ubuntu terbaru.

   Jalankan perintah berikut untuk memperbarui semua paket pada instans Anda.

   ```
   ubuntu:~$ sudo apt-get update
   ```

   ```
   ubuntu:~$ sudo apt-get upgrade
   ```

3. Instal paket EC2 instans Connect di instans Anda.

   ```
   ubuntu:~$ sudo apt-get install ec2-instance-connect
   ```

   Anda akan melihat tiga skrip baru di folder `/usr/share/ec2-instance-connect/`:

```
eic_curl_authorized_keys
eic_parse_authorized_keys
eic_run_authorized_keys
```

4. (Opsional) Memverifikasi bahwa EC2 Instance Connect berhasil diinstal pada instans Anda.

```
ubuntu:~$ sudo less /lib/systemd/system/ssh.service.d/ec2-instance-connect.conf
```

EC2 Instance Connect berhasil diinstal jika baris `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser` berisi nilai-nilai berikut:

```
AuthorizedKeysCommand /usr/share/ec2-instance-connect/eic_run_authorized_keys %
%u %%f
AuthorizedKeysCommandUser ec2-instance-connect
```

- `AuthorizedKeysCommand` mengatur skrip `eic_run_authorized_keys` untuk mencari kunci dari metadata instans
- `AuthorizedKeysCommandUser` menetapkan pengguna sistem sebagai `ec2-instance-connect`

> **Note**
>
> Jika sebelumnya Anda mengonfigurasi `AuthorizedKeysCommand` dan `AuthorizedKeysCommandUser`, instalasi EC2 Instance Connect tidak akan mengubah nilainya dan Anda tidak akan dapat menggunakan EC2 Instance Connect.

## Menghubungkan ke instans Linux menggunakan EC2 Instance Connect

Petunjuk berikut menjelaskan cara menyambung ke instans Linux menggunakan EC2 Instance Connect melalui konsol Amazon EC2 AWS CLI, atau klien SSH.

Saat Anda menyambung ke instans menggunakan EC2 Instance Connect melalui konsol atau AWS CLI, EC2 Instance Connect API secara otomatis mendorong kunci publik SSH ke metadata instans di mana kunci tersebut tetap ada selama 60 detik. Kebijakan IAM yang dilampirkan ke pengguna Anda mengizinkan tindakan ini. Jika Anda lebih suka menggunakan kunci SSH Anda

sendiri, Anda dapat menggunakan klien SSH dan secara eksplisit mendorong kunci SSH Anda ke instance menggunakan EC2 Instance Connect.

Pertimbangan-pertimbangan

Setelah menghubungkan ke instance menggunakan EC2 Instance Connect, koneksi tetap ada sampai sesi SSH dihentikan. Durasi koneksi tidak ditentukan oleh durasi kredenSIAL IAM Anda. Jika kredensi IAM Anda kedaluwarsa, koneksi akan tetap ada. Saat menggunakan pengalaman konsol EC2 Instance Connect, jika kredenSIAL IAM Anda kedaluwarsa, hentikan koneksi dengan menutup halaman browser. Saat menggunakan klien SSH Anda sendiri dan EC2 Instance Connect untuk menekan kunci Anda, Anda dapat mengatur nilai batas waktu SSH untuk mengakhiri sesi SSH secara otomatis.

Persyaratan

Sebelum Anda mulai, pastikan untuk meninjau prasyarat.

Opsi koneksi

- Hubungkan menggunakan konsol Amazon EC2
- Hubungkan menggunakan AWS CLI
- Hubungkan menggunakan kunci Anda sendiri dan klien SSH
- Pemecahan Masalah

Hubungkan menggunakan konsol Amazon EC2

Anda dapat menyambung ke instans menggunakan EC2 Instance Connect melalui konsol Amazon EC2.

Persyaratan

Untuk terhubung menggunakan konsol Amazon EC2, instans harus memiliki alamat IPv4 atau IPv6 publik. Jika instance hanya memiliki alamat IPv4 pribadi, Anda dapat menggunakan ec2-inst AWS CLI ance-connect untuk terhubung.

Untuk menyambung ke instans menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.

3. Pilih instans, lalu pilih Hubungkan.
4. Pilih tab EC2 Instance Connect.
5. Pilih  Hubungkan menggunakan IP Publik.
6. Jika ada pilihan, pilih alamat IP untuk terhubung. Jika tidak, alamat IP dipilih secara otomatis.
7. Untuk  Nama Pengguna, verifikasi nama pengguna.
8. Pilih  Connect  untuk membuat koneksi. Jendela terminal dalam peramban akan terbuka.

### Hubungkan menggunakan AWS CLI

Anda dapat menggunakan  ec2-instance-connect  AWS CLI untuk terhubung ke instance Anda dengan klien SSH. EC2 Instance Connect mencoba membuat koneksi menggunakan alamat IP yang tersedia dalam urutan yang telah ditentukan, berdasarkan jenis koneksi yang ditentukan. Jika alamat IP tidak tersedia, secara otomatis akan mencoba alamat berikutnya dalam urutan.

Tipe koneksi

`auto` (default)

EC2 Instance Connect mencoba menghubungkan menggunakan alamat IP instance dalam urutan berikut dan dengan jenis koneksi yang sesuai:

1. IPv4 publik: `direct`
2. IPv4 privat: `eice`
3. IPv6: `direct`

`direct`

EC2 Instance Connect mencoba menghubungkan menggunakan alamat IP instance dalam urutan berikut:

1. IPv4 Publik
2. IPv6
3. IPv4 pribadi (tidak terhubung melalui EC2 Instance Connect Endpoint)

`eice`

EC2 Instance Connect mencoba menghubungkan menggunakan alamat IPv4 pribadi instans dan Endpoint Connect Instance  EC2.

> **Note**
>
> Di masa depan, perilaku jenis `auto` koneksi mungkin berubah. Untuk memastikan bahwa jenis koneksi yang Anda inginkan digunakan, setel secara eksplisit `--connection-type` ke salah satu `direct` atau`eice`.

Persyaratan

Anda harus menggunakan AWS CLI versi 2. Untuk informasi selengkapnya, lihat Meng instal atau memperbarui ke versi terbaru AWS CLI.

Untuk terhubung ke instans menggunakan ID instans

Jika Anda hanya mengetahui ID instans, dan ingin membiarkan EC2 Instance Connect menentukan jenis koneksi yang akan digunakan saat menghubungkan ke instans Anda, gunakan perintah ec2-instance-connect ssh CLI dengan ID instance.

```
aws ec2-instance-connect ssh --instance-id i-1234567890example
```

Untuk menghubungkan ke instans menggunakan ID instans dan EC2 Instance Connect Endpoint

Jika Anda ingin terhubung ke instans Anda melalui EC2 Instance Connect Endpoint, gunakan perintah sebelumnya dan tentukan juga parameter `--connection-type` dengan nilai `eice`.

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --connection-type eice
```

Untuk terhubung ke instans menggunakan ID instans dan file kunci privat Anda sendiri

Jika Anda ingin terhubung ke instans Anda melalui Titik Akhir Connect instans EC2 menggunakan kunci pribadi Anda sendiri, tentukan ID instans dan jalur ke file kunci pribadi. Jangan sertakan *`file://`* di jalur; contoh berikut akan gagal:*`file:///path/to/key`*.

```
aws ec2-instance-connect ssh --instance-id i-1234567890example --private-key-file /
path/to/key.pem
```

> **Tip**
>
> Jika Anda mendapatkan kesalahan saat menggunakan perintah ini, pastikan Anda menggunakan AWS CLI versi 2, karena `ssh` perintah hanya tersedia di versi utama ini. Kami

juga merekomendasikan memperbarui secara teratur ke versi minor terbaru AWS CLI versi 2 untuk mengakses fitur-fitur terbaru. Untuk informasi selengkapnya, lihat Tentang AWS CLI versi 2 di Panduan AWS Command Line Interface Pengguna.

### Hubungkan menggunakan kunci Anda sendiri dan klien SSH

Anda dapat menggunakan kunci SSH Anda sendiri dan terhubung ke instans Anda dari klien SSH pilihan Anda jika menggunakan API EC2 Instance Connect. Ini memungkinkan Anda memanfaatkan kemampuan EC2 Instance Connect untuk mendorong kunci publik ke instans. Metode koneksi ini bekerja untuk instans dengan alamat IP privat dan publik.

#### Persyaratan

- Ketentuan untuk pasangan kunci
  - Tipe yang didukung: RSA (OpenSSH dan SSH2) dan ED25519
  - Panjang yang didukung: 2048 dan 4096.
  - Untuk informasi selengkapnya, lihat Membuat pasangan kunci menggunakan alat pihak ketiga dan mengimpor kunci publik ke Amazon EC2.
- Saat menghubungkan ke instance yang hanya memiliki alamat IP pribadi, komputer lokal tempat Anda memulai sesi SSH harus memiliki konektivitas ke titik akhir layanan EC2 Instance Connect (untuk mendorong kunci publik SSH Anda ke instans) serta konektivitas jaringan ke alamat IP pribadi instans untuk membuat sesi SSH. Titik akhir layanan EC2 Instance Connect dapat dijangkau melalui internet atau melalui antarmuka virtual Direct Connect publik. Untuk terhubung ke alamat IP privat instans, Anda dapat memanfaatkan layanan seperti Direct Connect, AWS Site-to-Site VPN, atau VPC peering.

#### Untuk terhubung ke instans Anda menggunakan kunci Anda sendiri dan klien SSH apa pun

1. (Opsional) Hasilkan kunci privat dan publik SSH baru

   Anda dapat menghasilkan kunci privat dan publik SSH baru, `my_key` dan `my_key.pub`, menggunakan perintah berikut:

   ```
   ssh-keygen -t rsa -f my_key
   ```

2. Dorong kunci publik SSH Anda ke instans

   Gunakan perintah send-ssh-public-key untuk mendorong kunci publik SSH Anda ke instans. Jika Anda meluncurkan instans menggunakan AL2023 atau Amazon Linux 2, nama pengguna default untuk AMI adalah`ec2-user`. Jika Anda meluncurkan instance Anda menggunakan Ubuntu, nama pengguna default untuk AMI adalah`ubuntu`.

   Contoh berikut mendorong kunci publik ke instans yang ditentukan di Zona Ketersediaan yang ditentukan, untuk mengautentikasi `ec2-user`.

   ```
   aws ec2-instance-connect send-ssh-public-key \
       --region us-west-2 \
       --availability-zone us-west-2b \
       --instance-id i-001234a4bf70dec41EXAMPLE \
       --instance-os-user ec2-user \
       --ssh-public-key file://my_key.pub
   ```

3. Hubungkan ke instans menggunakan kunci privat Anda

   Gunakan perintah ssh untuk terhubung ke instans menggunakan kunci privat sebelum kunci publik dihapus dari metadata instans (Anda memiliki waktu 60 detik sebelum kunci dihapus). Tentukan kunci pribadi yang sesuai dengan kunci publik, nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instans, dan nama DNS publik instans (jika terhubung melalui jaringan pribadi, tentukan nama DNS pribadi atau alamat IP). Tambahkan opsi `IdentitiesOnly=yes` untuk memastikan bahwa hanya file dalam konfigurasi ssh dan kunci tertentu yang digunakan untuk koneksi tersebut.

   ```
   ssh -o "IdentitiesOnly=yes" -i my_key ec2-
   user@ec2-198-51-100-1.compute-1.amazonaws.com
   ```

   Contoh berikut digunakan `timeout 3600` untuk mengatur sesi SSH Anda untuk berakhir setelah 1 jam. Proses yang dimulai selama sesi dapat terus berjalan di instans Anda setelah sesi berakhir.

   ```
   timeout 3600 ssh -o “IdentitiesOnly=yes” -i my_key ec2-
   user@ec2-198-51-100-1.compute-1.amazonaws.com
   ```

Pemecahan Masalah

Jika Anda menerima kesalahan saat mencoba untuk terhubung ke instans Anda, lihat hal berikut:

- Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux
- Bagaimana cara memecahkan masalah menghubungkan ke instans EC2 saya menggunakan EC2 instans Connect?

## Hapus instalasi EC2 Instance Connect

Untuk menonaktifkan EC2 Instance Connect, sambungkan ke instance Linux Anda dan hapus instalan `ec2-instance-connect` paket yang diinstal pada OS. Jika konfigurasi `sshd` cocok dengan pengaturan saat Anda menginstal EC2 Instance Connect, menghapus instalan `ec2-instance-connect` berarti juga menghapus konfigurasi `sshd`. Jika Anda memodifikasi konfigurasi `sshd` setelah menginstal EC2 Instance Connect, Anda harus memperbaruinya secara manual.

Amazon Linux

Anda dapat menghapus instalasi EC2 Instance Connect di AL2023 dan Amazon Linux 2 2.0.20190618 atau setelahnya, tempat EC2 Instance Connect telah dikonfigurasi sebelumnya.

Untuk menghapus instans EC2 Connect pada instans yang diluncurkan menggunakan Amazon Linux

1. Hubungkan ke instans Anda menggunakan SSH. Tentukan pasangan kunci SSH yang Anda gunakan untuk instance Anda saat Anda meluncurkannya dan nama pengguna default untuk AMI AL2023 atau Amazon Linux 2, yaitu`ec2-user`.

   Misalnya, perintah ssh berikut ini terhubung ke instans dengan nama DNS publik `ec2-a-b-c-d.us-west-2.compute.amazonaws.com`, menggunakan pasangan kunci `my_ec2_private_key.pem`.

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

2. Copot instalasi paket `ec2-instance-connect` menggunakan perintah yum.

   ```
   [ec2-user ~]$ sudo yum remove ec2-instance-connect
   ```

Ubuntu

Untuk menghapus instans EC2 Connect pada instance yang diluncurkan menggunakan AMI Ubuntu

1. Hubungkan ke instans Anda menggunakan SSH. Tentukan pasangan kunci SSH yang Anda gunakan untuk instance Anda ketika Anda meluncurkannya dan nama pengguna default untuk AMI Ubuntu, yaituubuntu.

   Misalnya, perintah ssh berikut ini terhubung ke instans dengan nama DNS publik `ec2-a-b-c-d.us-west-2.compute.amazonaws.com`, menggunakan pasangan kunci `my_ec2_private_key.pem`.

   ```
   $ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

2. Copot instalasi paket `ec2-instance-connect` menggunakan perintah apt-get.

   ```
   ubuntu:~$ sudo apt-get remove ec2-instance-connect
   ```

## Sambungkan ke instans Anda menggunakan alamat IP pribadi dan EC2 Instance Connect Endpoint

EC2 Instance Connect Endpoint memungkinkan Anda untuk terhubung dengan aman ke instans dari internet, tanpa menggunakan host bastion, atau mengharuskan cloud pribadi virtual (VPC) Anda memiliki konektivitas internet langsung.

Manfaat

- Anda dapat terhubung ke instans tanpa mengharuskan instans memiliki alamat IPv4 atau IPv6 publik. AWS biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastic. Untuk informasi selengkapnya, lihat tab Alamat IPv4 Publik di halaman harga Amazon VPC.
- Anda dapat terhubung ke instans Anda dari internet tanpa mengharuskan VPC Anda memiliki konektivitas internet langsung melalui gateway https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Internet_Gateway.html internet.
- Anda dapat mengontrol akses ke pembuatan dan penggunaan Titik Akhir Connect Instance EC2 untuk menyambung ke instans menggunakan kebijakan dan izin IAM.

- Semua upaya untuk terhubung ke instans Anda, baik yang berhasil maupun tidak berhasil, akan dicatat CloudTrail.

Harga

Tidak ada biaya tambahan untuk menggunakan EC2 Instance Connect Endpoints. Jika Anda menggunakan Endpoint Connect Instance EC2 untuk menyambung ke instans di Zona Ketersediaan yang berbeda, akan dikenakan biaya tambahan untuk transfer data di seluruh Availability Zone.

Daftar Isi


- Cara kerjanya
- Pertimbangan-pertimbangan
- Berikan izin untuk menggunakan EC2 Instance Connect Endpoint
- Grup keamanan untuk EC2 Instans Connect Endpoint
- Membuat EC2 Instance Connect Endpoint
- Memodifikasi Endpoint Connect Instance EC2
- Menghapus Titik Akhir Connect Instance EC2
- Menghubungkan ke instans Amazon EC2 menggunakan EC2 Instance Connect Endpoint
- Koneksi log dibuat melalui EC2 Instance Connect Endpoint
- Service-linked peran untuk EC2 Instance Connect Endpoint
- Kuota untuk Titik Akhir Connect Instance EC2

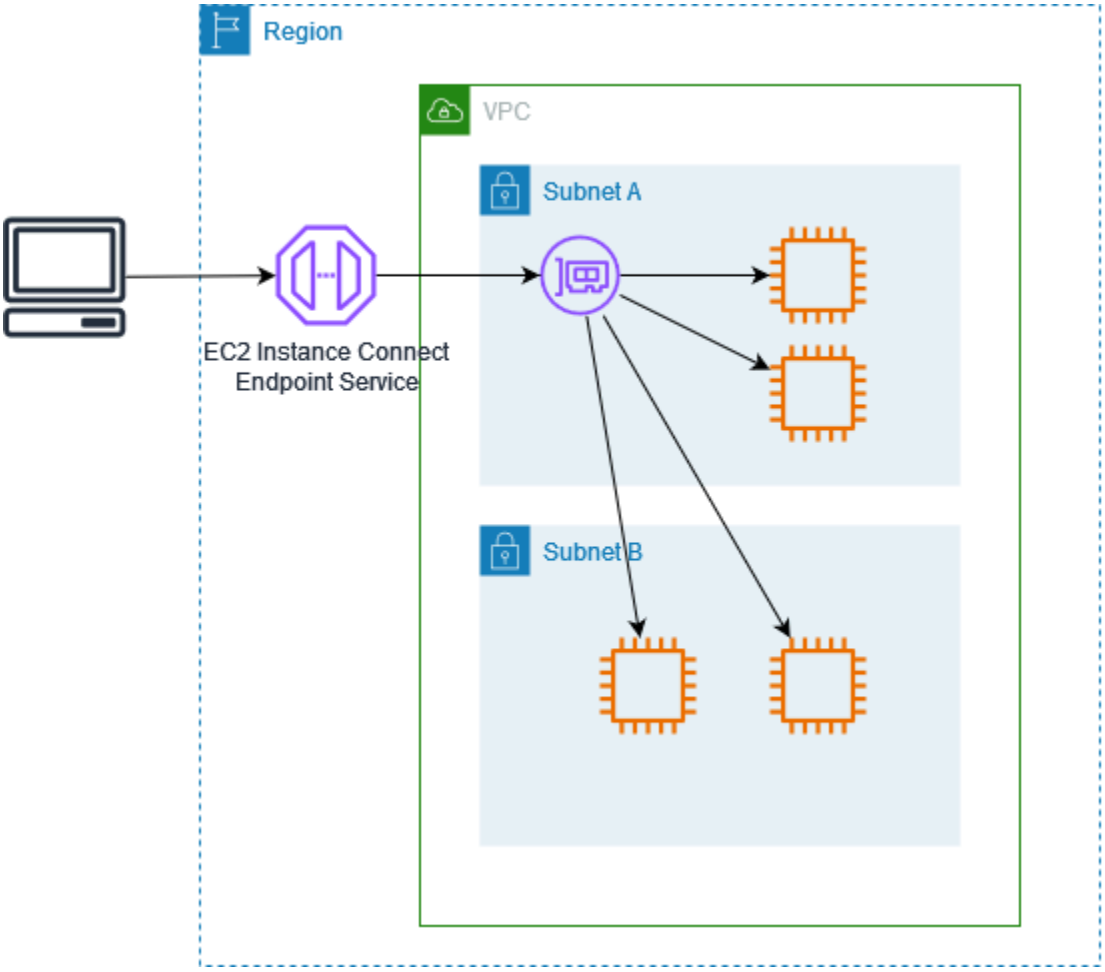

## Cara kerjanya

EC2 Instance Connect Endpoint adalah proxy TCP yang sadar identitas. Layanan Endpoint Connect Instance EC2 membuat terowongan pribadi dari komputer Anda ke titik akhir menggunakan kredensional untuk entitas IAM Anda. Lalu lintas diautentikasi dan diotorisasi sebelum mencapai VPC Anda.

Anda dapat mengonfigurasi aturan grup keamanan tambahan untuk membatasi lalu lintas masuk ke instans Anda. Misalnya, Anda dapat menggunakan aturan masuk untuk mengizinkan lalu lintas pada port manajemen hanya dari EC2 Instance Connect Endpoint.

Anda dapat mengonfigurasi aturan tabel rute untuk mengizinkan titik akhir terhubung ke instance apa pun di subnet VPC mana pun.

Diagram berikut menunjukkan bagaimana pengguna dapat terhubung ke instans mereka dari internet menggunakan EC2 Instance Connect Endpoint. Pertama, buat EC2 Instance Connect Endpoint di subnet A. Kami membuat antarmuka jaringan untuk titik akhir di subnet, yang berfungsi sebagai titik masuk untuk lalu lintas yang ditujukan ke instans Anda di VPC. Jika tabel rute untuk subnet B memungkinkan lalu lintas dari subnet A, maka Anda dapat menggunakan titik akhir untuk mencapai instance di subnet B.




## Pertimbangan-pertimbangan

Sebelum Anda mulai, pertimbangkan hal berikut.

- EC2 Instance Connect Endpoint ditujukan khusus untuk kasus penggunaan lalu lintas manajemen, bukan untuk transfer data volume tinggi. Data volume tinggi transfer dibatasi.
- Anda dapat membuat EC2 Instance Connect Endpoint untuk mendukung lalu lintas ke instans yang memiliki alamat IPv4 pribadi atau alamat IPv6. Jenis alamat IP dari titik akhir harus sesuai dengan alamat IP instance. Anda dapat membuat titik akhir yang mendukung semua jenis alamat IP.

- (Instans Linux) Jika Anda menggunakan pasangan kunci Anda sendiri, Anda dapat menggunakan AMI Linux apa pun. Jika tidak, instans Anda harus menginstal EC2 Instance Connect. Untuk informasi tentang AMI mana yang menyertakan EC2 Instance Connect dan cara menginstalnya di AMI lain yang didukung, lihatMenginstal EC2 Instance Connect.
- Anda dapat menetapkan grup keamanan ke Endpoint Connect Instance EC2. Jika tidak, kami menggunakan grup keamanan default untuk VPC. Grup keamanan untuk Endpoint Connect Instance EC2 harus mengizinkan lalu lintas keluar ke instans tujuan. Untuk informasi selengkapnya, lihat Grup keamanan untuk EC2 Instans Connect Endpoint.
- Anda dapat mengonfigurasi Endpoint Connect Instance EC2 untuk mempertahankan alamat IP sumber klien saat merutekan permintaan ke instans. Jika tidak, alamat IP antarmuka jaringan menjadi alamat IP klien untuk semua lalu lintas masuk.
  - Jika Anda mengaktifkan pelestarian IP klien, grup keamanan untuk instance harus mengizinkan lalu lintas dari klien. Selain itu, instans harus berada di VPC yang sama dengan EC2 Instance Connect Endpoint.
  - Jika Anda menonaktifkan pelestarian IP klien, grup keamanan untuk instance harus mengizinkan lalu lintas dari VPC. Ini adalah opsi default.
  - Pelestarian IP klien hanya didukung pada IPv4 EC2 Instance Connect Endpoints. Untuk menggunakan pelestarian IP klien, jenis alamat IP dari EC2 Instance Connect Endpoint harus IPv4. Pelestarian IP klien tidak didukung ketika jenis alamat IP adalah dual-stack atau IPv6.
  - Jenis instance berikut tidak mendukung pelestarian IP klien: C1, CG1, CG2, G1, HI1, M1, M2, M3, dan T1. Jika Anda mengaktifkan pelestarian IP klien dan mencoba menyambung ke instans dengan salah satu jenis instans ini dengan menggunakan EC2 Instance Connect Endpoint, koneksi gagal.
  - Pelestarian IP klien tidak didukung ketika lalu lintas dirutekan melalui gateway transit.
- Saat Anda membuat Endpoint Connect Instance EC2, peran terkait layanan secara otomatis dibuat untuk layanan Amazon EC2 di AWS Identity and Access Management (IAM). Amazon EC2 menggunakan peran tertaut layanan untuk menyediakan antarmuka jaringan di akun Anda, yang diperlukan saat membuat EC2 Instance Connect Endpoints. Untuk informasi selengkapnya, lihat Service-linked peran untuk EC2 Instance Connect Endpoint.
- Anda hanya dapat membuat 1 EC2 Instance Connect Endpoint per VPC dan per subnet. Untuk informasi selengkapnya, lihat Kuota untuk Titik Akhir Connect Instance EC2. Jika Anda perlu membuat Endpoint Connect Instance EC2 lain di Availability Zone yang berbeda dalam VPC yang sama, Anda harus terlebih dahulu menghapus Endpoint Connect Instance EC2 yang ada. Jika tidak, Anda akan menerima kesalahan kuota.

- Setiap EC2 Instance Connect Endpoint dapat mendukung hingga 20 koneksi bersamaan.
- Durasi maksimum untuk koneksi TCP yang mapan adalah 1 jam (3.600 detik). Anda dapat menentukan durasi maksimum yang diizinkan dalam kebijakan IAM, yang bisa mencapai 3.600 detik. Untuk informasi selengkapnya, lihat Izin untuk menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans.

  Durasi koneksi tidak ditentukan oleh durasi kredenSIAL IAM Anda. Jika kredensi IAM Anda kedaluwarsa, koneksi akan terus bertahan hingga durasi maksimum yang ditentukan tercapai. Saat Anda menyambung ke instans menggunakan pengalaman konsol EC2 Instance Connect Endpoint, setel Durasi terowongan maksimum (detik) ke nilai yang kurang dari durasi kredentif IAM Anda. Jika kredentif IAM Anda kedaluwarsa lebih awal, hentikan koneksi ke instans Anda dengan menutup halaman browser.

## Berikan izin untuk menggunakan EC2 Instance Connect Endpoint

Secara default, entitas IAM tidak memiliki izin untuk membuat, mendeskripsikan, atau memodifikasi EC2 Instance Connect Endpoints. Administrator IAM dapat membuat kebijakan IAM yang memberikan izin yang diperlukan untuk melakukan tindakan tertentu pada sumber daya yang mereka butuhkan.

Untuk informasi tentang pembuatan kebijakan IAM, lihat Membuat kebijakan IAM di Panduan Pengguna IAM.

Contoh kebijakan berikut menunjukkan bagaimana Anda dapat mengontrol izin yang dimiliki pengguna untuk EC2 Instance Connect Endpoints.

Contoh

- Izin untuk membuat, mendeskripsikan, memodifikasi, dan menghapus EC2 Instance Connect Endpoints
- Izin untuk menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans
- Izin untuk terhubung hanya dari rentang alamat IP tertentu

### Izin untuk membuat, mendeskripsikan, memodifikasi, dan menghapus EC2 Instance Connect Endpoints

Untuk membuat dan memodifikasi Endpoint Connect Instance EC2, pengguna memerlukan izin untuk tindakan berikut:

- `ec2:CreateInstanceConnectEndpoint`
- `ec2:CreateNetworkInterface`
- `ec2:CreateTags`
- `ec2:ModifyInstanceConnectEndpoint`
- `iam:CreateServiceLinkedRole`

Untuk mendeskripsikan dan menghapus EC2 Instance Connect Endpoint, pengguna memerlukan izin untuk tindakan berikut:

- `ec2:DescribeInstanceConnectEndpoints`
- `ec2:DeleteInstanceConnectEndpoint`

Anda dapat membuat kebijakan yang memberikan izin untuk membuat, menjelaskan, memodifikasi, dan menghapus EC2 Instance Connect Endpoint di semua subnet. Atau, Anda dapat membatasi tindakan untuk subnet tertentu hanya dengan menentukan ARN subnet sebagai `Resource` yang diizinkan atau dengan menggunakan kunci syarat `ec2:SubnetID`. Anda juga dapat menggunakan kunci syarat `aws:ResourceTag` untuk secara eksplisit mengizinkan atau menolak pembuatan titik akhir dengan tanda tertentu. Untuk informasi selengkapnya, lihat Kebijakan dan izin di IAM pada Panduan Pengguna IAM.

Contoh kebijakan IAM

Dalam contoh kebijakan IAM berikut, `Resource` bagian tersebut memberikan izin untuk membuat, memodifikasi, dan menghapus titik akhir di semua subnet, yang ditentukan oleh tanda bintang (). * Tindakan API `ec2:Describe*` tidak mendukung izin tingkat sumber daya. Oleh karena itu, wildcard * dibutuhkan dalam `Resource`.

Izin untuk menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans

Tindakan `ec2-instance-connect:OpenTunnel` tersebut memberikan izin untuk membuat koneksi TCP ke instans untuk terhubung melalui EC2 Instance Connect Endpoint. Anda dapat menentukan EC2 Instance Connect Endpoint untuk digunakan. Atau, `Resource` dengan tanda bintang (*) memungkinkan pengguna untuk menggunakan semua EC2 Instance Connect Endpoint yang tersedia. Anda juga dapat membatasi akses ke instans berdasarkan ada atau tidak adanya tanda sumber daya sebagai kunci syarat.

Kondisi

- `ec2-instance-connect:remotePort`- Port pada instance yang dapat digunakan untuk membuat koneksi TCP. Ketika kunci syarat ini digunakan, percobaan untuk terhubung ke instans pada port lain selain port yang ditentukan dalam kebijakan akan mengakibatkan kegagalan.
- `ec2-instance-connect:privateIpAddress`Alamat IP pribadi tujuan yang terkait dengan instance yang ingin Anda buat koneksi TCP. Anda dapat menentukan satu alamat IP, seperti `10.0.0.1/32`, atau rentang IP melalui CIDR, seperti `10.0.1.0/28`. Ketika kunci syarat ini digunakan, percobaan untuk terhubung ke sebuah instans dengan alamat IP privat yang berbeda atau di luar rentang CIDR akan mengakibatkan kegagalan.
- `ec2-instance-connect:maxTunnelDuration`— Durasi maksimum untuk koneksi TCP yang mapan. Satuannya adalah detik dan durasinya berkisar dari minimal 1 detik hingga maksimum 3.600 detik (1 jam). Jika kondisi tidak ditentukan, durasi default diatur ke 3.600 detik (1 jam). Mencoba terhubung ke instans lebih lama dari durasi yang ditentukan dalam kebijakan IAM atau lebih lama dari maksimum default akan menghasilkan kegagalan. Koneksi terputus setelah durasi yang ditentukan.

  Jika `maxTunnelDuration` ditentukan dalam kebijakan IAM dan nilai yang ditentukan kurang dari 3.600 detik (default), maka Anda harus menentukan `--max-tunnel-duration` dalam perintah saat menghubungkan ke sebuah instans. Untuk informasi selengkapnya tentang terhubung ke instans Anda, lihat Menghubungkan ke instans Amazon EC2 menggunakan EC2 Instance Connect Endpoint.

Anda juga dapat memberikan akses pengguna untuk membuat koneksi ke instans berdasarkan keberadaan tag resource pada Endpoint Connect Instance EC2. Untuk informasi selengkapnya, lihat Kebijakan dan izin di IAM dalam Panduan Pengguna IAM.

Untuk instance Linux, `ec2-instance-connect:SendSSHPublicKey` tindakan memberikan izin untuk mendorong kunci publik ke instance. Kondisi `ec2:osuser` menentukan nama pengguna OS (sistem operasi) yang dapat mendorong kunci publik ke sebuah instans. Gunakan nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instance. Untuk informasi selengkapnya, lihat Berikan izin IAM untuk EC2 instans Connect.

Contoh kebijakan IAM

Contoh kebijakan IAM berikut mengizinkan prinsipal IAM untuk terhubung ke instans hanya menggunakan Endpoint Connect Instance EC2 yang ditentukan, yang diidentifikasi oleh ID titik

akhir yang ditentukan. `eice-123456789abcdef` Koneksi berhasil dibuat hanya jika semua kondisi terpenuhi.

> **Note**
>
> Tindakan API `ec2:Describe*` tidak mendukung izin tingkat sumber daya. Oleh karena itu, wildcard `*` dibutuhkan dalam `Resource`.

Linux

Contoh ini mengevaluasi apakah koneksi ke instance dibuat pada —port 22 (SSH), jika alamat IP pribadi instance terletak dalam kisaran `10.0.1.0/31` (antara `10.0.1.0` dan`10.0.1.1`), dan `maxTunnelDuration` kurang dari atau sama dengan `3600` detik. Koneksi terputus setelah `3600` detik (1 jam).

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Sid": "EC2InstanceConnect",
            "Action": "ec2-instance-connect:OpenTunnel",
            "Effect": "Allow",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:instance-connect-
endpoint/eice-123456789abcdef",
            "Condition": {
                "NumericEquals": {
                    "ec2-instance-connect:remotePort": "22"
                },
                "IpAddress": {
                    "ec2-instance-connect:privateIpAddress": "10.0.1.0/31"
                },
                "NumericLessThanEquals": {
                    "ec2-instance-connect:maxTunnelDuration": "3600"
                }
            }
        },
        {
            "Sid": "SSHPublicKey",
            "Effect": "Allow",
```

```
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:osuser": "ami-username"
                }
            }
        },
        {
            "Sid": "Describe",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceConnectEndpoints"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Windows

Contoh ini mengevaluasi apakah koneksi ke instance dibuat pada port 3389 (RDP), jika alamat IP pribadi instance terletak dalam kisaran `10.0.1.0/31` (antara `10.0.1.0` dan`10.0.1.1`), dan kurang dari atau sama dengan detik. `maxTunnelDuration` `3600` Koneksi terputus setelah `3600` detik (1 jam).

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Sid": "EC2InstanceConnect",
            "Action": "ec2-instance-connect:OpenTunnel",
            "Effect": "Allow",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:instance-connect-
endpoint/eice-123456789abcdef",
            "Condition": {
                "NumericEquals": {
                    "ec2-instance-connect:remotePort": "3389"
                },
```

```
                "IpAddress": {
                    "ec2-instance-connect:privateIpAddress": "10.0.1.0/31"
                },
                "NumericLessThanEquals": {
                    "ec2-instance-connect:maxTunnelDuration": "3600"
                }
            }
        },
        {
            "Sid": "Describe",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceConnectEndpoints"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

Izin untuk terhubung hanya dari rentang alamat IP tertentu

Contoh kebijakan IAM berikut memungkinkan pengguna utama IAM untuk terhubung ke instans dengan syarat instans tersebut terhubung dari alamat IP dalam rentang alamat IP yang ditentukan dalam kebijakan. Jika utama IAM memanggil `OpenTunnel` dari alamat IP yang tidak berada di dalam `192.0.2.0/24` (contoh rentang alamat IP dalam kebijakan ini), responsnya adalah`Access Denied`. Untuk informasi selengkapnya, lihat `aws:SourceIp` dalam Panduan Pengguna IAM.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "ec2-instance-connect:OpenTunnel",
            "Resource": "arn:aws:ec2:us-east-1:111122223333:instance-connect-
endpoint/eice-123456789abcdef",
            "Condition": {
                "IpAddress": {
                    "aws:SourceIp": "192.0.2.0/24"
                },
```

```
                "NumericEquals": {
                    "ec2-instance-connect:remotePort": "22"
                }
            }
        },
        {
            "Sid": "SSHPublicKey",
            "Effect": "Allow",
            "Action": "ec2-instance-connect:SendSSHPublicKey",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "ec2:osuser": "ami-username"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceConnectEndpoints"
            ],
            "Resource": "*"
        }
    ]
}
```

## Grup keamanan untuk EC2 Instans Connect Endpoint

Grup keamanan mengontrol lalu lintas yang diizinkan untuk menjangkau dan meninggalkan sumber daya yang terkait dengannya. Misalnya, kami menolak lalu lintas ke dan dari instans Amazon EC2 kecuali jika secara khusus diizinkan oleh grup keamanan yang terkait dengan instans.

Contoh berikut menunjukkan cara mengonfigurasi aturan grup keamanan untuk EC2 Instance Connect Endpoint dan instans target.

Contoh

- Aturan grup keamanan EC2 Instance Connect Endpoint
- Aturan grup keamanan instans target

### Aturan grup keamanan EC2 Instance Connect Endpoint

Aturan grup keamanan untuk Endpoint Connect Instance EC2 harus mengizinkan lalu lintas keluar yang ditujukan untuk instans target meninggalkan titik akhir. Anda dapat menentukan grup keamanan instans atau rentang alamat IPv4 atau IPv6 dari VPC sebagai tujuan.

Lalu lintas ke titik akhir berasal dari Layanan Endpoint Connect Instance EC2, dan diizinkan terlepas dari aturan masuk untuk grup keamanan titik akhir. Untuk mengontrol siapa yang dapat menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans, gunakan kebijakan IAM. Untuk informasi selengkapnya, lihat Izin untuk menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans.

#### Contoh aturan keluar: Referensi grup keamanan

Contoh berikut menggunakan referensi grup keamanan, yang berarti bahwa tujuan adalah grup keamanan yang terkait dengan instance target. Aturan ini memungkinkan lalu lintas keluar dari titik akhir ke semua instance yang menggunakan grup keamanan ini.

| Protokol | Tujuan | Rentang port | Komentar |
|---|---|---|---|
| TCP | *ID of instance security group* | 22 | Mengizinkan lalu lintas SSH keluar ke semua instans yang terkait dengan grup keamanan instance |

#### Contoh aturan keluar: rentang alamat IPv4

Contoh berikut memungkinkan lalu lintas keluar ke rentang alamat IPv4 yang ditentukan. Alamat IPv4 dari sebuah instance ditetapkan dari subnetnya, sehingga Anda dapat menggunakan rentang alamat IPv4 dari VPC.

| Protokol | Tujuan | Rentang port | Komentar |
|---|---|---|---|
| TCP | *VPC IPv4 CIDR* | 22 | Memungkinkan lalu lintas SSH keluar ke VPC |

#### Contoh aturan keluar: rentang alamat IPv6

Contoh berikut memungkinkan lalu lintas keluar ke rentang alamat IPv6 yang ditentukan. Alamat IPv6 dari sebuah instance ditetapkan dari subnetnya, sehingga Anda dapat menggunakan rentang alamat IPv6 dari VPC.

| Protokol | Tujuan | Rentang port | Komentar |
| --- | --- | --- | --- |
| TCP | *VPC IPv6 CIDR* | 22 | Memungkinkan lalu lintas SSH keluar ke VPC |

Aturan grup keamanan instans target

Aturan grup keamanan untuk instans target harus mengizinkan lalu lintas masuk dari EC2 Instance Connect Endpoint. Anda dapat menentukan grup keamanan titik akhir atau rentang alamat IPv4 atau IPv6 sebagai sumbernya. Jika Anda menentukan rentang alamat IPv4, sumbernya bergantung pada apakah pelestarian IP klien dinonaktifkan atau aktif. Untuk informasi selengkapnya, lihat Pertimbangan-pertimbangan.

Karena grup keamanan bersifat stateful, lalu lintas respons diizinkan meninggalkan VPC terlepas dari aturan keluar untuk grup keamanan instans.

Contoh aturan masuk: Referensi grup keamanan

Contoh berikut menggunakan referensi grup keamanan, yang berarti bahwa sumbernya adalah grup keamanan yang terkait dengan titik akhir. Aturan ini memungkinkan lalu lintas SSH masuk dari titik akhir ke semua instance yang menggunakan grup keamanan ini, apakah pelestarian IP klien aktif atau nonaktif. Jika tidak ada aturan grup keamanan masuk lainnya untuk SSH, maka instance menerima lalu lintas SSH hanya dari titik akhir.

| Protokol | Sumber | Rentang port | Komentar |
| --- | --- | --- | --- |
| TCP | *ID of endpoint security group* | 22 | Mengizinkan lalu lintas SSH masuk dari sumber daya yang terkait dengan grup keamanan titik akhir |

Contoh aturan masuk: Pelestarian IP klien mati

Contoh berikut memungkinkan lalu lintas SSH masuk dari rentang alamat IPv4 yang ditentukan. Karena pelestarian IP klien tidak aktif, alamat IPv4 sumber adalah alamat antarmuka jaringan titik akhir. Alamat antarmuka jaringan titik akhir ditetapkan dari subnetnya, sehingga Anda dapat menggunakan rentang alamat IPv4 dari VPC untuk mengizinkan koneksi ke semua instance di VPC.

| Protokol | Sumber | Rentang port | Komentar |
| --- | --- | --- | --- |
| TCP | *VPC IPv4 CIDR* | 22 | Memungkinkan lalu lintas SSH masuk dari VPC |

Contoh aturan masuk: Pelestarian IP klien aktif

Contoh berikut memungkinkan lalu lintas SSH masuk dari rentang alamat IPv4 yang ditentukan. Karena pelestarian IP klien aktif, alamat IPv4 sumber adalah alamat klien.

| Protokol | Sumber | Rentang port | Komentar |
| --- | --- | --- | --- |
| TCP | *Public IPv4 address range* | 22 | Mengizinkan lalu lintas masuk dari rentang alamat IPv4 klien yang ditentukan |

## Membuat EC2 Instance Connect Endpoint

Anda dapat membuat EC2 Instance Connect Endpoint untuk memungkinkan koneksi aman ke instans Anda.

Pertimbangan-pertimbangan

- Subnet bersama — Anda dapat membuat EC2 Instance Connect Endpoint di subnet yang dibagikan dengan Anda. Namun, Anda tidak dapat menggunakan EC2 Instance Connect Endpoints yang dibuat pemilik VPC di subnet yang dibagikan dengan Anda.
- Jenis alamat IP — EC2 Instance Connect Endpoints mendukung jenis alamat berikut, yang harus kompatibel dengan subnet Anda:
  - `ipv4`— Hubungkan hanya ke instans EC2 dengan alamat IPv4 pribadi.
  - `dualstack`— Hubungkan ke instans EC2 dengan alamat IPv4 pribadi atau alamat IPv6.
  - `ipv6`— Sambungkan hanya ke instans EC2 dengan alamat IPv6.

Prasyarat

Anda harus memiliki izin IAM yang diperlukan untuk membuat EC2 Instance Connect Endpoint. Untuk informasi selengkapnya, lihat Izin untuk membuat, mendeskripsikan, memodifikasi, dan menghapus EC2 Instance Connect Endpoints.

Console

Untuk membuat EC2 Instance Connect Endpoint

1. Buka konsol Amazon VPC di https://console.aws.amazon.com/vpc/.
2. Pada panel navigasi kiri, pilih Titik Akhir.
3. Pilih Buat titik akhir, lalu tentukan pengaturan titik akhir sebagai berikut:
   a. (Opsional) Untuk Tanda nama, masukkan nama untuk titik akhir.
   b. Untuk Jenis, pilih EC2 Instance Connect Endpoint.
   c. Di bawah Pengaturan jaringan, untuk VPC, pilih VPC yang memiliki instans target.
   d. (Opsional) Untuk mempertahankan alamat IP klien, perluas Peng aturan tambahan dan pilih kotak cent ang Pertahankan IP Klien. Jika tidak, defaultnya adalah menggunakan antarmuka jaringan titik akhir sebagai alamat IP klien.

      > **Note**
      > Opsi ini hanya tersedia bila jenis alamat IP titik akhir dikonfigurasi sebagai IPv4.

   e. (Opsional) Untuk Grup keamanan, pilih grup keamanan untuk dikaitkan dengan titik akhir. Jika tidak, defaultnya adalah menggunakan grup keamanan default untuk VPC. Untuk informasi selengkapnya, lihat Grup keamanan untuk EC2 Instans Connect Endpoint.
   f. Untuk Subnet, pilih subnet untuk membuat titik akhir.
   g. Untuk jenis alamat IP, pilih jenis alamat IP untuk titik akhir. Pilih Dualstack jika Anda perlu mendukung koneksi IPv4 dan IPv6 ke instans Anda. Pilih IPv4 jika Anda perlu mendukung pelestarian IP klien.
   h. (Opsional) Untuk menambahkan tanda, pilih Tambahkan tanda baru dan masukkan kunci dan nilai tanda.
4. Tinjau pengaturan Anda, lalu pilih Buat titik akhir.

Status awal titik akhir adalah Pen ding. Sebelum Anda dapat menyambung ke instance menggunakan titik akhir ini, Anda harus menunggu hingga status titik akhir tersedia. Hal ini dapat menghabiskan waktu beberapa menit.

5. Untuk menyambung ke instance menggunakan titik akhir Anda, lihatSambungkan ke instans.

AWS CLI

Untuk membuat EC2 Instance Connect Endpoint

Gunakan perintah create-instance-connect-endpoint.

```
aws ec2 create-instance-connect-endpoint \
    --subnet-id subnet-0123456789example
```

Untuk menentukan jenis lalu lintas yang didukung titik akhir, sertakan `--ip-address-type` parameter. Nilai yang valid adalah `ipv4`, `dualstack`, atau `ipv6`. Subnet harus mendukung jenis alamat IP yang Anda tentukan. Ketika `--ip-address-type` parameter dihilangkan, nilai default ditentukan oleh jenis alamat IP yang didukung oleh subnet.

```
aws ec2 create-instance-connect-endpoint \
    --subnet-id subnet-0123456789example \
    --ip-address-type ipv4
```

Berikut ini adalah output contoh.

```
{
        "OwnerId": "111111111111",
        "InstanceConnectEndpointId": "eice-0123456789example",
        "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
        "State": "create-complete",
        "StateMessage": "",
        "DnsName": "eice-0123456789example.0123abcd.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
        "FipsDnsName": "eice-0123456789example.0123abcd.fips.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com",
        "NetworkInterfaceIds": [
            "eni-0123abcd"
        ],
        "VpcId": "vpc-0123abcd",
```

```
        "AvailabilityZone": "us-east-1a",
        "AvailabilityZoneId": "use1-az4",
        "CreatedAt": "2023-04-07T15:43:53.000Z",
        "SubnetId": "subnet-0123abcd",
        "PreserveClientIp": false,
        "SecurityGroupIds": [
            "sg-0123abcd"
        ],
        "Tags": [],
        "IpAddressType": "ipv4"
}
```

Untuk memantau status pembuatan

Nilai awal untuk bidang `State` adalah `create-in-progress`. Sebelum Anda dapat terhubung ke instans menggunakan titik akhir ini, tunggu sampai statusnya `create-complete`. Gunakan describe-instance-connect-endpoints perintah untuk memantau status EC2 Instance Connect Endpoint. `--query`Parameter menyaring hasil ke `State` bidang.

```
aws ec2 describe-instance-connect-endpoints --instance-connect-endpoint-
ids eice-0123456789example --query InstanceConnectEndpoints[*].State --output text
```

Berikut ini adalah output contoh.

```
create-complete
```

PowerShell

Untuk membuat EC2 Instance Connect Endpoint

Gunakan New-EC2InstanceConnectEndpoint cmdlet.

```
New-EC2InstanceConnectEndpoint -SubnetId subnet-0123456789example
```

Untuk menentukan jenis lalu lintas yang didukung titik akhir, sertakan `-IpAddressType` parameter. Nilai yang valid adalah `ipv4`, `dualstack`, atau `ipv6`. Subnet harus mendukung jenis alamat IP yang Anda tentukan. Ketika `-IpAddressType` parameter dihilangkan, nilai default ditentukan oleh jenis alamat IP yang didukung oleh subnet.

```
New-EC2InstanceConnectEndpoint -SubnetId subnet-0123456789example -
IpAddressType ipv4
```

Berikut ini adalah output contoh.

```
OwnerId                     : 111111111111
InstanceConnectEndpointId   : eice-0123456789example
InstanceConnectEndpointArn  : arn:aws:ec2:us-east-1:111111111111:instance-connect-
endpoint/eice-0123456789example
State                       : create-complete
StateMessage                :
DnsName                     : eice-0123456789example.0123abcd.ec2-instance-connect-
endpoint.us-east-1.amazonaws.com
FipsDnsName                 : eice-0123456789example.0123abcd.fips.ec2-instance-
connect-endpoint.us-east-1.amazonaws.com
NetworkInterfaceIds         : {eni-0123abcd}
VpcId                       : vpc-0123abcd
AvailabilityZone            : us-east-1a
AvailabilityZoneId          : use1-az4
CreatedAt                   : 4/7/2023 3:43:53 PM
SubnetId                    : subnet-0123abcd
PreserveClientIp            : False
SecurityGroupIds            : {sg-0123abcd}
Tags                        : {}
IpAddressType               : ipv4
```

Untuk memantau status pembuatan

Nilai awal untuk bidang `State` adalah `create-in-progress`. Sebelum Anda dapat terhubung ke instans menggunakan titik akhir ini, tunggu sampai statusnya `create-complete`. Gunakan Get-EC2InstanceConnectEndpoint cmdlet untuk memantau status EC2 Instance Connect Endpoint. `.State.Value`menyaring hasil ke `State` bidang.

```
(Get-EC2InstanceConnectEndpoint -InstanceConnectEndpointId
 "eice-0123456789example").State.Value
```

Berikut ini adalah output contoh.

```
create-complete
```

## Memodifikasi Endpoint Connect Instance EC2

Anda dapat memodifikasi EC2 Instance Connect Endpoint yang ada menggunakan konsol, AWS CLI, atau SDK.

Sebelum memulai, Anda harus memiliki izin IAM yang diperlukan. Untuk informasi selengkapnya, lihat Izin untuk membuat, mendeskripsikan, memodifikasi, dan menghapus EC2 Instance Connect Endpoints.

Parameter yang dapat Anda modifikasi

Anda dapat mengubah parameter EC2 Instance Connect Endpoint berikut ini:

Grup keamanan

Anda dapat menentukan grup keamanan baru untuk EC2 Instance Connect Endpoint. Grup keamanan baru menggantikan grup keamanan saat ini.

Saat memodifikasi grup keamanan, Anda harus menentukan:

- Setidaknya satu grup keamanan, meskipun itu hanya grup keamanan default di VPC.
- ID grup keamanan, bukan nama.

Jenis alamat IP

Anda dapat menentukan jenis alamat IP baru untuk EC2 Instance Connect Endpoint.

Nilai yang valid: `ipv4` | `dualstack` | `ipv6`

Pertahankan pengaturan IP klien

Anda dapat menentukan apakah akan mempertahankan alamat IP klien sebagai sumber.

> **Note**
>
> Mempertahankan IP klien hanya didukung pada IPv4 EC2 Instance Connect Endpoints. Saat mengaktifkan`PreserveClientIp`, jenis alamat IP titik akhir yang ada harus`ipv4`, atau jika memodifikasi jenis alamat IP dalam permintaan yang sama, nilai baru harus`ipv4`.

Console

Untuk memodifikasi Endpoint Connect Instance EC2

1. Buka konsol Amazon VPC di https://console.aws.amazon.com/vpc/.
2. Pada panel navigasi kiri, pilih Titik Akhir.

3. Pilih titik akhir.
4. Pilih T ind akan, Ubah pengaturan titik akhir.
5. Ubah pengaturan titik akhir.
6. Pilih Simpan perubahan.

   Titik akhir memasuki status Pending. Ketika modifikasi selesai, titik akhir kembali ke status Tersedia.

AWS CLI

Untuk memodifikasi Endpoint Connect Instance EC2

Gunakan modify-instance-connect-endpoint perintah dan tentukan EC2 Instance Connect Endpoint dan parameter yang akan dimodifikasi. Contoh berikut memodifikasi semua parameter dalam satu permintaan.

```
aws ec2 modify-instance-connect-endpoint \
    --instance-connect-endpoint-id eice-0123456789example \
    --security-group-ids sg-0123456789example \
    --ip-address-type dualstack \
    --no-preserve-client-ip
```

Berikut ini adalah output contoh.

```
{
    "Return": true
}
```

Untuk memantau status pembaruan

Selama modifikasi, status EC2 Instance Connect Endpoint berubah menjadi`update-in-progress`. Proses pembaruan berjalan secara asinkron dan selesai dengan status `update-complete` atau`update-failed`. Titik akhir menggunakan konfigurasi lamanya sampai status berubah menjadi`update-complete`.

Gunakan describe-instance-connect-endpoints perintah untuk memantau status pembaruan. `--query`Parameter menyaring hasil ke `State` bidang.

```
aws ec2 describe-instance-connect-endpoints \
```

```
    --instance-connect-endpoint-ids eice-0123456789example \
    --query InstanceConnectEndpoints[*].State --output text
```

Berikut ini adalah output contoh.

```
update-complete
```

PowerShell

Untuk memodifikasi Endpoint Connect Instance EC2

Gunakan Edit-EC2InstanceConnectEndpoint cmdlet dan tentukan EC2 Instance Connect Endpoint dan parameter yang akan dimodifikasi. Contoh berikut memodifikasi semua parameter dalam satu permintaan.

```
Edit-EC2InstanceConnectEndpoint `
    -InstanceConnectEndpointId eice-0123456789example `
    -SecurityGroupIds sg-0123456789example `
    -IpAddressType dualstack `
    -PreserveClientIp $false
```

Berikut ini adalah output contoh.

```
True
```

Untuk memantau status pembaruan

Selama modifikasi, status EC2 Instance Connect Endpoint berubah menjadi`update-in-progress`. Proses pembaruan berjalan secara asinkron dan selesai dengan status `update-complete` atau`update-failed`. Titik akhir menggunakan konfigurasi lamanya sampai status berubah menjadi`update-complete`.

Gunakan Get-EC2InstanceConnectEndpoint perintah untuk memantau status pembaruan. `.State.Value`menyaring hasil ke `State` bidang.

```
(Get-EC2InstanceConnectEndpoint -InstanceConnectEndpointId
 "eice-0123456789example").State.Value
```

Berikut ini adalah output contoh.

```
update-complete
```

## Menghapus Titik Akhir Connect Instance EC2

Setelah selesai menggunakan EC2 Instance Connect Endpoint, Anda dapat menghapusnya.

Anda harus memiliki izin IAM yang diperlukan untuk membuat EC2 Instance Connect Endpoint. Untuk informasi selengkapnya, lihat Izin untuk membuat, mendeskripsikan, memodifikasi, dan menghapus EC2 Instance Connect Endpoints.

Saat Anda menghapus EC2 Instance Connect Endpoint menggunakan konsol, itu memasuki status Menghapus. Jika penghapusan berhasil, titik akhir yang dihapus tidak lagi muncul. Jika penghapusan gagal, status pengh apus-gagal dan pesan Status memberikan alasan kegagalan.

Saat Anda menghapus Endpoint Connect Instance EC2 menggunakan AWS CLI, titik akhir akan memasuki `delete-in-progress` status. Jika penghapusan berhasil, ia memasuki `delete-complete` status. Jika penghapusan gagal, status adalah `delete-failed` dan `StateMessage` memberikan alasan kegagalan.

Console

Untuk menghapus EC2 Instance Connect Endpoint

1. Buka konsol Amazon VPC di https://console.aws.amazon.com/vpc/.
2. Pada panel navigasi kiri, pilih Titik Akhir.
3. Pilih titik akhir.
4. Pilih Tindakan, Hapus titik akhir VPC.
5. Saat diminta mengonfirmasi, pilih **`delete`**.
6. Pilih Hapus.

AWS CLI

Untuk menghapus EC2 Instance Connect Endpoint

Gunakan delete-instance-connect-endpoint perintah dan tentukan ID dari EC2 Instance Connect Endpoint untuk dihapus.

```
aws ec2 delete-instance-connect-endpoint --instance-connect-endpoint-
id eice-03f5e49b83924bbc7
```

Berikut ini adalah output contoh.

```
{
    "InstanceConnectEndpoint": {
        "OwnerId": "111111111111",
        "InstanceConnectEndpointId": "eice-0123456789example",
        "InstanceConnectEndpointArn": "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example",
        "State": "delete-in-progress",
        "StateMessage": "",
        "NetworkInterfaceIds": [],
        "VpcId": "vpc-0123abcd",
        "AvailabilityZone": "us-east-1d",
        "AvailabilityZoneId": "use1-az2",
        "CreatedAt": "2023-02-07T12:05:37+00:00",
        "SubnetId": "subnet-0123abcd"
    }
}
```

PowerShell

Untuk menghapus EC2 Instance Connect Endpoint

Gunakan Remove-EC2InstanceConnectEndpoint cmdlet dan tentukan ID Endpoint Connect Instance EC2 yang akan dihapus.

```
Remove-EC2InstanceConnectEndpoint -InstanceConnectEndpointId eice-03f5e49b83924bbc7
```

Berikut ini adalah output contoh.

```
@{
    InstanceConnectEndpoint = @{
        OwnerId = "111111111111"
        InstanceConnectEndpointId = "eice-0123456789example"
        InstanceConnectEndpointArn = "arn:aws:ec2:us-east-1:111111111111:instance-
connect-endpoint/eice-0123456789example"
        State = "delete-in-progress"
        StateMessage = ""
```

```
        NetworkInterfaceIds = @()
        VpcId = "vpc-0123abcd"
        AvailabilityZone = "us-east-1d"
        AvailabilityZoneId = "use1-az2"
        CreatedAt = "2023-02-07T12:05:37+00:00"
        SubnetId = "subnet-0123abcd"
    }
}
```

## Menghubungkan ke instans Amazon EC2 menggunakan EC2 Instance Connect Endpoint

Anda dapat menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans Amazon EC2 yang mendukung SSH atau RDP.

Prasyarat

- Anda harus memiliki izin IAM yang diperlukan untuk terhubung ke EC2 Instance Connect Endpoint. Untuk informasi selengkapnya, lihat Izin untuk menggunakan EC2 Instance Connect Endpoint untuk menyambung ke instans.
- Endpoint Connect Instance EC2 harus berada di salah satu status berikut:
  - buat-lengkap untuk titik akhir baru
  - pembaruan dalam proses, pembaru an-selesai, atau pembaruan gagal untuk titik akhir yang ada sedang dimodifikasi. Saat memodifikasi titik akhir, itu terus menggunakan konfigurasi aslinya sampai status berubah menjadi pembaru an-selesai.

    Jika VPC Anda tidak memiliki EC2 Instance Connect Endpoint, Anda dapat membuatnya. Untuk informasi selengkapnya, lihat Membuat EC2 Instance Connect Endpoint.
- Jenis alamat IP EC2 Instance Connect Endpoint harus kompatibel dengan jenis alamat IP instans. Jika tipe alamat IP endpoint Anda adalah dual-stack, maka itu dapat berfungsi untuk alamat IPv4 dan IPv6.
- (Instans Linux) Untuk menggunakan konsol Amazon EC2 untuk menyambung ke instans Anda, atau untuk menggunakan CLI untuk menghubungkan dan EC2 Instance Connect menangani kunci sementara, instans Anda harus menginstal EC2 Instance Connect. Untuk informasi selengkapnya, lihat Menginstal EC2 Instance Connect.
- Pastikan grup keamanan instans mengizinkan lalu lintas SSH masuk dari EC2 Instance Connect Endpoint. Untuk informasi selengkapnya, lihat Aturan grup keamanan instans target.

Opsi koneksi

- Hubungkan ke instans Linux Anda menggunakan konsol Amazon EC2
- Hubungkan ke instans Linux Anda, menggunakan SSH.
- Hubungkan ke instance Linux Anda dengan ID instansnya menggunakan AWS CLI
- Hubungkan ke instans Windows Anda menggunakan RDP
- Pemecahan Masalah

Hubungkan ke instans Linux Anda menggunakan konsol Amazon EC2

Anda dapat menyambung ke instance menggunakan konsol Amazon EC2 (klien berbasis browser) sebagai berikut.

Untuk menyambung ke instans menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Hubun gkan.
4. Pilih tab EC2 Instance Connect.
5. Untuk Jenis koneksi, pilih Hu bungkan menggunakan IP Pribadi.
6. Pilih alamat IPv4 pribadi atau alamat IPv6. Opsi tersedia berdasarkan alamat IP yang ditetapkan ke instance Anda. Jika opsi berwarna abu-abu, instance Anda tidak memiliki alamat IP jenis yang ditetapkan untuknya.
7. Untuk Endpoint Connect Instance EC2, pilih ID Endpoint Connect Instance EC2.

> **Note**
>
> EC2 Instance Connect Endpoint harus kompatibel dengan alamat IP yang Anda pilih pada langkah sebelumnya. Jika tipe alamat IP endpoint Anda adalah dual-stack, maka itu dapat berfungsi untuk alamat IPv4 dan IPv6. Untuk informasi selengkapnya, lihat Membuat EC2 Instance Connect Endpoint.

8. Untuk Nama Pengguna, jika AMI yang Anda gunakan untuk meluncurkan instance menggunakan nama pengguna selain`ec2-user`, masukkan nama pengguna yang benar.
9. Untuk durasi terowongan Max (detik), masukkan durasi maksimum yang diizinkan untuk koneksi SSH.

Durasi harus sesuai dengan `maxTunnelDuration` kondisi apa pun yang ditentukan dalam kebijakan IAM. Jika Anda tidak memiliki akses ke kebijakan IAM, hubungi administrator Anda.

10. Pilih Hubungkan. Ini membuka jendela terminal untuk contoh Anda.

Hubungkan ke instans Linux Anda, menggunakan SSH.

Anda dapat menggunakan SSH untuk terhubung ke instans Linux Anda, dan menggunakan perintah `open-tunnel` untuk membuat terowongan privat. Anda dapat menggunakan `open-tunnel` dalam mode koneksi tunggal atau multikoneksi. Anda dapat menentukan ID instans, alamat IPv4 pribadi, atau alamat IPv6.

Untuk informasi tentang menggunakan AWS CLI untuk menyambung ke instance Anda menggunakan SSH, lihatHubungkan menggunakan AWS CLI.

Contoh berikut menggunakan OpenSSH. Anda dapat menggunakan klien SSH lain yang mendukung mode proxy.

Koneksi tunggal

Untuk mengizinkan hanya satu koneksi ke instans menggunakan SSH dan perintah **`open-tunnel`**

Gunakan `ssh` dan open-tunnel AWS CLI perintah sebagai berikut. Perintah `-o` proxy menyertakan `open-tunnel` perintah yang membuat terowongan pribadi ke instans.

```
ssh -i my-key-pair.pem ec2-user@i-1234567890abcdef0 \
    -o ProxyCommand='aws ec2-instance-connect open-tunnel --instance-
id i-1234567890abcdef0'
```

Untuk:

- `-i` – Menentukan pasangan kunci yang digunakan untuk meluncurkan instans.
- *`ec2-user`*`@`*`i-1234567890abcdef0`* – Menentukan nama pengguna AMI yang digunakan untuk meluncurkan instans dan ID instans. Untuk instance dengan alamat IPv6, Anda harus menentukan alamat IPv6 alih-alih ID instance.
- `--instance-id` – Menentukan ID instans yang akan dihubungkan. Atau, tentukan `%h`, yang mengekstrak ID instans dari pengguna. Untuk contoh dengan alamat IPv6, ganti `--instance-id` *`i-1234567890abcdef0`* dengan`--private-ip-address` *`2001:db8::1234:5678:1.2.3.4`*.

Multi-connection

Untuk mengizinkan beberapa koneksi ke instance, pertama-tama jalankan open-tunnel AWS CLI perintah untuk mulai mendengarkan koneksi TCP baru, lalu gunakan `ssh` untuk membuat koneksi TCP baru dan terowongan pribadi ke instance Anda.

Untuk mengizinkan beberapa koneksi ke instance Anda menggunakan SSH dan perintah **`open-tunnel`**

1. Jalankan perintah berikut untuk mulai mendengarkan koneksi TCP baru pada port yang ditentukan pada komputer lokal Anda.

   ```
   aws ec2-instance-connect open-tunnel \
       --instance-id i-1234567890abcdef0 \
       --local-port 8888
   ```

   Keluaran yang diharapkan

   ```
   Listening for connections on port 8888.
   ```

2. Di jendela terminal baru, jalankan perintah `ssh` berikut untuk membuat koneksi TCP baru dan terowongan pribadi ke instans Anda.

   ```
   ssh -i my-key-pair.pem ec2-user@localhost -p 8888
   ```

   Output yang diharapkan — Di jendela terminal pertama, Anda akan melihat hal-hal berikut:

   ```
   [1] Accepted new tcp connection, opening websocket tunnel.
   ```

   Anda mungkin juga melihat yang berikut ini:

   ```
   [1] Closing tcp connection.
   ```

Hubungkan ke instance Linux Anda dengan ID instansnya menggunakan AWS CLI

Jika Anda hanya mengetahui ID instans Anda, Anda dapat menggunakan AWS CLI perintah ec2-instance-connect ssh untuk menyambung ke instance Anda menggunakan klien SSH. Untuk informasi selengkapnya, lihat Hubungkan menggunakan AWS CLI.

Prasyarat

- Inst AWS CLI al versi 2 dan konfigurasikan menggunakan kredenSIAL Anda. Untuk informasi selengkapnya, lihat Meng instal atau memperbarui ke versi terbaru AWS CLI dan Meng onfigurasi AWS CLI di Panduan AWS Command Line Interface Pengguna.
- Atau, buka AWS CloudShell dan jalankan AWS CLI perintah di shell pra-autentikasi.

Untuk menghubungkan ke instans menggunakan ID instans dan EC2 Instance Connect Endpoint

Jika Anda hanya mengetahui ID instance, gunakan perintah ec2-instance-connect ssh CLI, dan tentukan perintah, ID instance, dan `--connection-type` parameter dengan `eice` nilai untuk menggunakan EC2 Instance Connect Endpoint. `ssh` Jika instance hanya memiliki alamat IPv6, Anda juga harus menyertakan `--instance-ip` parameter dengan alamat IPv6.

- Jika instance memiliki alamat IPv4 pribadi (juga dapat memiliki alamat IPv6) gunakan perintah dan parameter berikut:

```
aws ec2-instance-connect ssh \
    --instance-id i-1234567890example \
    --os-user ec2-user \
    --connection-type eice
```

- Jika instance hanya memiliki alamat IPv6, sertakan `--instance-ip` parameter dengan alamat IPv6:

```
aws ec2-instance-connect ssh \
    --instance-id i-1234567890example \
    --instance-ip 2001:db8::1234:5678:1.2.3.4 \
    --os-user ec2-user \
    --connection-type eice
```

> **Tip**
>
> Jika Anda mendapatkan kesalahan, pastikan Anda menggunakan AWS CLI versi 2. `ssh`Parameter hanya tersedia di AWS CLI versi 2. Untuk informasi selengkapnya, lihat Tentang AWS CLI versi 2 di Panduan AWS Command Line Interface Pengguna.

Hubungkan ke instans Windows Anda menggunakan RDP

Anda dapat menggunakan Remote Desktop Protocol (RDP) melalui EC2 Instance Connect Endpoint untuk terhubung ke instans Windows tanpa alamat IPv4 publik atau nama DNS publik.

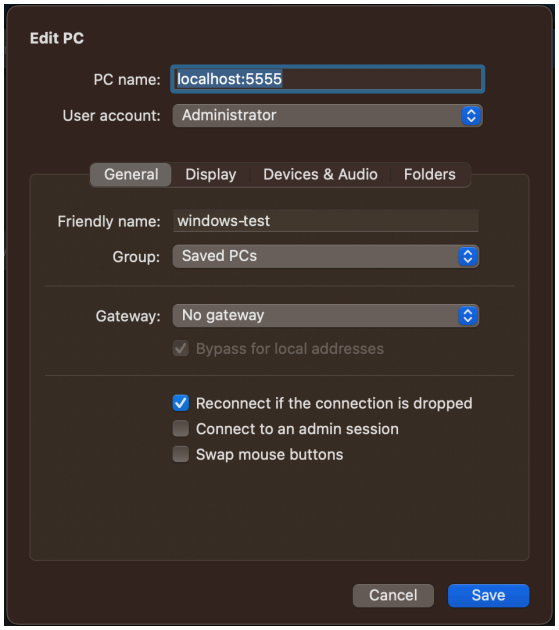
Untuk terhubung ke instans Windows menggunakan klien RDP

1. Selesaikan Langkah 1 - 8 di Hu bungkan ke instance Windows Anda menggunakan RDP. Setelah mengunduh file desktop RDP pada Langkah 8, Anda akan mendapatkan pesan Unable to connect, yang diharapkan karena instance Anda tidak memiliki alamat IP publik.
2. Jalankan perintah berikut untuk membuat terowongan privat ke VPC di mana instans berada. `--remote-port` harus `3389` karena RDP menggunakan port 3389 secara default.

   ```
   aws ec2-instance-connect open-tunnel \
       --instance-id i-1234567890abcdef0 \
       --remote-port 3389 \
       --local-port any-port
   ```

3. Di folder Unduhan Anda, temukan file desktop RDP yang Anda unduh, dan seret ke jendela klien RDP.
4. Buka menu konteks (klik kanan) untuk file desktop RDP dan pilih Edit.
5. Di jendela Edit PC, untuk nama PC (instance yang akan dihubungkan), masukkan`localhost:local-port`, di mana *`local-port`* menggunakan nilai yang sama seperti yang Anda tentukan di Langkah 2, lalu pilih S impan.

   Perhatikan bahwa tangkapan layar berikut dari jendela Edit PC berasal dari Microsoft Remote Desktop di Mac. Jika Anda menggunakan klien Windows, jendelanya mungkin berbeda.




6. Di klien RDP, buka menu konteks (klik kanan) untuk PC (yang baru saja Anda konfigurasikan) dan pilih Hubun gkan untuk terhubung ke instance Anda.
7. Pada saat diminta, masukkan kata sandi terdekripsi untuk akun administrator.

## Pemecahan Masalah

Gunakan informasi berikut untuk membantu mendiagnosis dan memperbaiki masalah yang mungkin Anda temukan saat menggunakan EC2 instans Connect Endpoint untuk menghubungkan sebuah instans.

### Tidak dapat terhubung ke instans Anda

Berikut ini adalah alasan umum mengapa Anda mungkin tidak dapat terhubung ke instans Anda.

- Grup keamanan — Periksa grup keamanan yang ditetapkan ke EC2 Instance Connect Endpoint dan instans Anda. Untuk informasi selengkapnya tentang aturan grup keamanan yang diperlukan, lihat Grup keamanan untuk EC2 Instans Connect Endpoint.
- Status instans - Verifikasi apakah instans Anda ada dalam `running` status.
- Pasangan kunci - Jika perintah yang Anda gunakan untuk menghubungkan memerlukan kunci pribadi, verifikasi bahwa instans Anda memiliki kunci publik dan Anda memiliki kunci pribadi yang sesuai.
- Izin IAM - Verifikasi apakah Anda memiliki izin IAM yang diperlukan. Untuk informasi selengkapnya, lihat Berikan izin untuk menggunakan EC2 Instance Connect Endpoint.

Untuk kiat pemecahan masalah selengkapnya untuk instans Linux, lihatMemecahkan masalah saat menyambung ke instans Amazon EC2 Linux. Untuk tips pemecahan masalah untuk instans Windows, lihatthe section called “Masalah RDP instance Windows”.

ErrorCode: AccessDeniedException

Jika Anda menerima `AccessDeniedException` kesalahan, dan `maxTunnelDuration` kondisinya ditentukan dalam kebijakan IAM, pastikan untuk menentukan `--max-tunnel-duration` parameter saat menghubungkan ke sebuah instans. Untuk informasi selengkapnya tentang parameter ini, lihat open-tunnel di Referensi Perintah AWS CLI .

## Koneksi log dibuat melalui EC2 Instance Connect Endpoint

Anda dapat mencatat operasi sumber daya dan mengaudit koneksi yang dibuat melalui EC2 Instance Connect Endpoint dengan AWS CloudTrail log.

Untuk informasi selengkapnya tentang penggunaan AWS CloudTrail dengan Amazon EC2, lihatLog panggilan API Amazon EC2 menggunakan AWS CloudTrail.

Log Instans EC2 Hubungkan panggilan API Endpoint dengan AWS CloudTrail

Operasi sumber daya EC2 Instance Connect Endpoint dicatat CloudTrail sebagai peristiwa manajemen. Ketika panggilan API berikut dilakukan, aktivitas direkam sebagai CloudTrail peristiwa di ri wayat acara:

- `CreateInstanceConnectEndpoint`
- `DescribeInstanceConnectEndpoints`
- `DeleteInstanceConnectEndpoint`

Anda dapat melihat, mencari, dan mengunduh acara terbaru di Akun AWS. Untuk informasi selengkapnya, lihat Melihat CloudTrail acara dengan riwayat acara di Panduan AWS CloudTrail Pengguna.

Gunakan AWS CloudTrail untuk mengaudit pengguna yang terhubung ke instans menggunakan EC2 Instance Connect Endpoint

Upaya koneksi ke instans melalui EC2 Instance Connect Endpoint dicatat CloudTrail di riwayat peristiwa. Ketika koneksi ke instans dimulai melalui EC2 Instance Connect Endpoint, koneksi dicatat sebagai peristiwa CloudTrail manajemen dengan `eventName` of`OpenTunnel`.

Anda dapat membuat EventBridge aturan Amazon yang merut CloudTrail ekan acara ke target. Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

Berikut ini adalah contoh `OpenTunnel` peristiwa manajemen yang masuk CloudTrail.


```
{
     "eventVersion": "1.08",
     "userIdentity": {
         "type": "IAMUser",
         "principalId": "ABCDEFGONGNOMOOCB6XYTQEXAMPLE",
         "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
         "accountId": "123456789012",
         "accessKeyId": "ABCDEFGUKZHNAW4OSN2AEXAMPLE",
         "userName": "IAM-friendly-name"
     },
     "eventTime": "2023-04-11T23:50:40Z",
     "eventSource": "ec2-instance-connect.amazonaws.com",
     "eventName": "OpenTunnel",
     "awsRegion": "us-east-1",
     "sourceIPAddress": "1.2.3.4",
     "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
     "requestParameters": {
         "instanceConnectEndpointId": "eici-0123456789EXAMPLE",
         "maxTunnelDuration": "3600",
         "remotePort": "22",
         "privateIpAddress": "10.0.1.1"
     },
     "responseElements": null,
     "requestID": "98deb2c6-3b3a-437c-a680-03c4207b6650",
     "eventID": "bbba272c-8777-43ad-91f6-c4ab1c7f96fd",
     "readOnly": false,
     "resources": [{
         "accountId": "123456789012",
         "type": "AWS::EC2::InstanceConnectEndpoint",
         "ARN": "arn:aws:ec2:us-east-1:123456789012:instance-connect-endpoint/
eici-0123456789EXAMPLE"
     }],
     "eventType": "AwsApiCall",
     "managementEvent": true,
     "recipientAccountId": "123456789012",
     "eventCategory": "Management"
}
```

## Service-linked peran untuk EC2 Instance Connect Endpoint

Amazon EC2 menggunakan peran terkait AWS Identity and Access Management layanan (IAM). Peran terkait layanan adalah jenis peran IAM unik yang ditautkan langsung ke Amazon EC2. Service-linked peran telah ditentukan sebelumnya oleh Amazon EC2 dan menyertakan semua izin yang diperlukan Amazon EC2 untuk memanggil orang lain Layanan AWS atas nama Anda. Untuk informasi selengkapnya, lihat Service-linked peran di Panduan Pengguna IAM.

Service-linked izin peran untuk EC2 Instance Connect Endpoint

Amazon EC2 digunakan AWSServiceRoleForEC2InstanceConnect untuk membuat dan mengelola antarmuka jaringan di akun Anda yang diperlukan oleh EC2 Instance Connect Endpoint.

Peran terkait layanan AWSServiceRoleForEC2InstanceConnect memercayai layanan berikut untuk mengambil peran tersebut:

- `ec2-instance-connect.amazonaws.com`

Per AWSServiceRoleForEC2InstanceConnect an terkait layanan menggunakan kebijakan terkelola berikut:

- Ec2InstanceConnectEndpoint

Untuk melihat izin untuk kebijakan terkelola, lihat Ec2InstanceConnectEndpoint di Referensi Kebijakan AWS Terkelola.

Membuat peran terkait layanan untuk EC2 Instance Connect Endpoint

Anda tidak perlu membuat peran tertaut layanan secara manual. Saat Anda membuat EC2 Instance Connect Endpoint, Amazon EC2 membuat peran terkait layanan untuk Anda.

Mengedit peran terkait layanan untuk EC2 Instance Connect Endpoint

EC2 Instance Connect Endpoint tidak mengizinkan Anda mengedit peran terkait AWSServiceRoleForEC2InstanceConnect layanan.

Menghapus peran terkait layanan untuk EC2 Instance Connect Endpoint

Jika Anda tidak perlu lagi menggunakan EC2 Instance Connect Endpoint, sebaiknya hapus peran terkait AWSServiceRoleForEC2InstanceConnect layanan.

Anda harus menghapus semua sumber daya EC2 Instance Connect Endpoint sebelum dapat menghapus peran terkait layanan.

Untuk menghapus peran terkait layanan, lihat Menghapus peran terkait layanan di Panduan Pengguna IAM.

Anda harus mengonfigurasi izin untuk mengizinkan entitas IAM (pengguna, grup, atau peran) membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat izin Service-linked peran di Panduan Pengguna IAM.

### Kuota untuk Titik Akhir Connect Instance EC2

Anda Akun AWS memiliki kuota default, sebelumnya disebut sebagai batas, untuk setiap AWS layanan. Kecuali dinyatakan lain, setiap kuota adalah Region-specific.

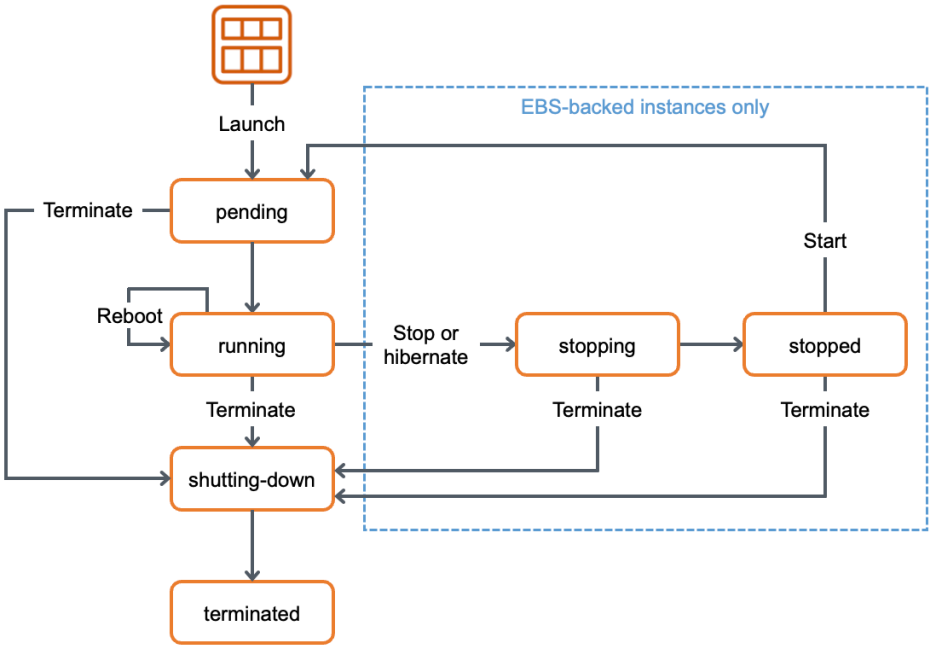
Anda Akun AWS memiliki kuota berikut terkait dengan EC2 Instance Connect Endpoint.

| Nama | Default | Dapat disesuaikan |
|---|---|---|
| Jumlah maksimum Titik Akhir Connect Instans EC2 per per Akun AWS Wilayah AWS | 5 | Tidak |
| Jumlah Maksimum EC2 instans Connect per VPC | 1 | Tidak |
| Jumlah maksimum EC2 instans Connect Endpoint per subnet | 1 | Tidak |
| Jumlah maksimum koneksi bersamaan per EC2 Instance Connect Endpoint | 20 | Tidak |

## Perubahan status instans Amazon EC2

Instans Amazon EC2 bertransisi melalui status yang berbeda dari saat Anda meluncurkannya hingga pengakhirannya.

Ilustrasi berikut menunjukkan transisi di antara status instans.




Anda dapat menerima pemberitahuan saat instans berubah status. Untuk informasi selengkapnya, lihat the section called “Peristiwa perubahan status”.

## Penagihan berdasarkan status instans

Tabel berikut memberikan deskripsi singkat tentang setiap status instans dan menunjukkan apakah penggunaan instance ditagih. Beberapa AWS sumber daya, seperti volume Amazon EBS dan alamat IP Elastic, dikenakan biaya terlepas dari status instans. Untuk informasi selengkapnya, lihat Menghindari Biaya Tidak Terduga dalam AWS Billing Panduan Pengguna .

| Status instans | Deskripsi | Penagihan penggunaan instans |
| --- | --- | --- |
| `pendinc` | Instans sedang bersiap untuk memasuki status `running`. Sebuah instans memasuki status `pending` saat diluncurkan atau ketika dimulai setelah berada dalam status `stopped`. | Tidak ditagih |

| Status instans | Deskripsi | Penagihan penggunaan instans |
|---|---|---|
| `running` | Instans ini sedang berjalan dan siap digunakan. | Dikenakan biaya |
| `stopping` | Instans sedang bersiap untuk dihentikan. | Tidak ditagih<br><br>**Note**<br>Jika Anda melakukan hibernasi instans, Anda akan ditagih saat instans berada dalam status. `stopping` |
| `stopped` | Instans ini dimatikan dan tidak dapat digunakan. Instans ini dapat dimulai kapan saja. | Tidak ditagih |
| `shutting-down` | Instans sedang bersiap untuk diakhiri. | Tidak ditagih |
| `terminated` | Instans ini telah dihapus secara permanen dan tidak dapat dimulai. | Tidak ditagih<br><br>**Note**<br>Instans Terpesan yang diterapkan ke instans yang diakhiri akan ditagih hingga akhir jangka waktunya sesuai dengan opsi pembayaran mereka. Untuk informasi selengkapnya, lihat Ikhtisar Instans Cadangan untuk Amazon EC2 |

## Instans yang tertunda

Saat Anda meluncurkan sebuah instans, instans akan memasuki ststua `pending`. Tipe instans yang Anda tentukan saat peluncuran menentukan perangkat keras komputer host untuk instans Anda.

Amazon Machine Image (AMI) yang Anda tentukan saat peluncuran digunakan untuk mem-boot instance. Setelah instans siap untuk Anda, instans memasuki status `running`. Anda dapat terhubung ke instans yang sedang berjalan dan menggunakannya seperti Anda menggunakan komputer yang ada di depan Anda.

Segera setelah instans Anda bertransisi ke status `running`, Anda akan dikenai biaya untuk setiap detik Anda menjalankan instans, dengan minimum satu menit, meskipun instans tetap idle dan Anda tidak terhubung dengannya.

## Instans yang dihentikan

Jika instans Anda gagal dalam pemeriksaan status atau tidak menjalankan aplikasi Anda seperti yang diharapkan, dan jika volume root instans Anda adalah volume Amazon EBS, Anda dapat menghentikan dan memulai instans Anda untuk mencoba memperbaiki masalah.

Saat Anda menghentikan sebuah instans, instans tersebut akan memasuki status `stopping`, lalu status `stopped`. Anda tidak dikenakan biaya untuk penggunaan atau biaya transfer data untuk instans Anda yang sedang `stopped`. Biaya dikenakan untuk penyimpanan volume Amazon EBS apa pun. Saat instans Anda ada dalam status `stopped`, Anda dapat memodifikasi atribut tertentu dari instans, termasuk tipe instans.

Saat Anda memulai instans Anda, instans memasuki status `pending`, dan instans akan dipindahkan ke komputer host baru (meskipun dalam beberapa kasus, instant tetap di host saat ini). Saat Anda menghentikan dan memulai instans, Anda kehilangan data apa pun di volume penyimpanan instans yang dilampirkan ke komputer host sebelumnya.

Instans Anda mempertahankan alamat IPv4 privatnya, yang berarti bahwa alamat IP Elastis yang terkait dengan alamat IPv4 privat atau antarmuka jaringan tetap terkait dengan instans Anda. Jika instans Anda memiliki alamat IPv6, maka instans tersebut mempertahankan alamat IPv6.

Setiap kali Anda melakukan transisi atas sebuah instans dari `stopped` ke `running`, Anda akan dikenai biaya per detik ketika instans sedang berjalan, dengan minimal satu menit setiap kali instans dimulai.

Untuk detail selengkapnya tentang penghentian dan pemulaian sebuah instans, lihat Menghentikan dan memulai instans Amazon EC2.

## Instans hibernasi

Saat Anda melakukan hibernasi instance, sistem operasi diberi sinyal untuk melakukan hibernasi (suspend-to-disk), yang menyimpan konten dari memori instans (RAM) ke volume root Amazon EBS Anda. Volume root Amazon EBS instans dan volume data Amazon EBS terlampir akan tetap ada. Saat Anda memulai instans, volume root Amazon EBS dipulihkan ke keadaan sebelumnya dan konten RAM dimuat ulang. Volume data terlampir sebelumnya akan dilampirkan kembali dan instans akan mempertahankan ID instansnya.

Saat Anda menghentikan sebuah instans, instans tersebut akan memasuki status `stopping`, lalu status `stopped`. Anda tidak dikenakan biaya untuk penggunaan instans hibernasi saat berada di `stopped` status, tetapi Anda dikenakan biaya saat berada di `stopping` status, tidak seperti ketika Anda menghentikan instance tanpa hibernasi. Anda tidak dikenakan biaya transfer data, tetapi Anda dikenakan biaya untuk penyimpanan untuk volume Amazon EBS apa pun, termasuk penyimpanan untuk data RAM.

Ketika Anda memulai instance hibernasi Anda, instans memasuki `pending` status, dan instance dipindahkan ke komputer host baru (meskipun dalam beberapa kasus, tetap pada host saat ini).

Instans Anda mempertahankan alamat IPv4 privatnya, yang berarti bahwa alamat IP Elastis yang terkait dengan alamat IPv4 privat atau antarmuka jaringan masih terkait dengan instans Anda. Jika instans Anda memiliki alamat IPv6, maka instans tersebut mempertahankan alamat IPv6-nya.

Untuk informasi selengkapnya, lihat Hibernasi instans Amazon EC2 Anda.

## Mem-boot ulang instance

Anda dapat melakukan boot ulang instans Anda menggunakan konsol Amazon EC2, alat baris perintah, dan API Amazon EC2. Kami menyarankan agar Anda menggunakan Amazon EC2 untuk melakukan boot ulang instans Anda alih-alih menjalankan perintah boot ulang sistem operasi dari instans Anda.

Mem-boot ulang sebuah instans sama dengan mem-boot ulang sistem operasi. Instans tetap berada di komputer host yang sama dan mempertahankan nama DNS publiknya, alamat IP privat, dan data apa pun pada volume penyimpanan instansnya. Biasanya diperlukan waktu beberapa menit untuk menyelesaikan booting ulang, tetapi waktu yang diperlukan untuk memulai ulang bergantung pada konfigurasi instans.

Mem-boot ulang sebuah instans tidak memulai periode penagihan instans baru; penagihan per detik berlanjut tanpa biaya minimum satu menit lebih lanjut.

Untuk informasi selengkapnya, lihat Mulai ulang instans Amazon EC2 Anda.

## Instans yang dihentikan

Jika Anda telah memutuskan bahwa Anda tidak lagi membutuhkan sebuah instans, Anda dapat mengakhirinya. Segera setelah status instans berubah menjadi `shutting-down` atau `terminated`, Anda akan berhenti dikenai biaya untuk instans tersebut.

Jika Anda mengaktifkan perlindungan pengakhiran, Anda tidak dapat mengakhiri instans menggunakan konsol, CLI, atau API.

Setelah Anda mengakhiri sebuah instans, instans akan tetap terlihat di konsol untuk beberapa saat, kemudian entri tersebut akan dihapus secara otomatis. Anda juga dapat mendeskripsikan instans yang diakhiri menggunakan CLI dan API. Sumber daya (seperti tanda) secara bertahap dipisahkan dari instans yang diakhiri, oleh karena itu mungkin tidak lagi terlihat pada instans yang diakhiri setelah beberapa saat. Anda tidak dapat terhubung ke atau memulihkan instans yang diakhiri.

Setiap EBS-backed instans Amazon mendukung `InstanceInitiatedShutdownBehavior` atribut, yang mengontrol apakah instance berhenti atau dihentikan saat Anda memulai shutdown dari dalam instans itu sendiri (misalnya, dengan menggunakan shutdown perintah di Linux). Perilaku defaultnya adalah menghentikan instans. Anda dapat memodifikasi pengaturan atribut ini saat instans sedang berjalan atau berhenti.

Setiap volume Amazon EBS mendukung atribut `DeleteOnTermination`, yang mengontrol apakah volume dihapus atau dipertahankan saat Anda menghentikan instans tempatnya dilampirkan. Defaultnya adalah menghapus volume root dan mempertahankan volume EBS lainnya.

Untuk informasi selengkapnya, lihat Menghentikan instans Amazon EC2.

## Perbedaan antara status instance

Tabel berikut merangkum perbedaan utama antara me-reboot, menghentikan, hibernasi, dan menghentikan instans Anda.

| Karakteristik | Boot ulang | Stop/start (Hanya EBS-backed instans Amazon) | Hibernasi (hanya EBS-backed instans Amazon) | Mengakhiri |
|---|---|---|---|---|
| Komputer host | Instance tetap di komputer host yang sama. | Instance dipindahkan ke komputer host baru (meskipun dalam beberapa kasus, tetap pada host saat ini). | Instance dipindahkan ke komputer host baru (meskipun dalam beberapa kasus, tetap pada host saat ini). | Tidak ada |
| Alamat IPv4 privat | Instans tetap mempertahankan alamat IPv4 privatnya. | Instans tetap mempertahankan alamat IPv4 privatnya. | Instans tetap mempertahankan alamat IPv4 privatnya. | Tidak ada |
| Alamat IPv4 publik | Instans menyimpan alamat IPv4 publiknya. | Instans mendapatkan alamat IPv4 publik baru, kecuali jika memiliki antarmuka jaringan sekunder atau alamat IPv4 pribadi sekunder yang terkait dengan alamat IP Elastic. | Instans mendapatkan alamat IPv4 publik baru, kecuali jika memiliki antarmuka jaringan sekunder atau alamat IPv4 pribadi sekunder yang terkait dengan alamat IP Elastic. | Tidak ada |
| Alamat Elastic IP (IPv4) | Alamat IP Elastis tetap terkait dengan instans | Alamat IP Elastis tetap terkait dengan instans | Alamat IP Elastis tetap terkait dengan instans | Alamat IP Elastis dipisahkan dari instans |
| Alamat IPv6 | Instans tetap mempertahankan alamat IPv6-nya | Instans tetap mempertahankan alamat IPv6-nya | Instans tetap mempertahankan alamat IPv6-nya | Tidak ada |

| Karakteristik | Boot ulang | Stop/start (Hanya EBS-backed instans Amazon) | Hibernasi (hanya EBS-backed instans Amazon) | Mengakhiri |
|---|---|---|---|---|
| Volume toko instan | Data disimpan | Datanya dihapus | Datanya dihapus | Datanya dihapus |
| Volume root | Volume dipertahankan | Volume dipertahankan | Volume dipertahankan | Volume dihapus secara default |
| RAM (isi memori) | RAM dihapus | RAM dihapus | RAM disimpan ke file di volume root | RAM dihapus |
| Penagihan | Jam penagihan instans tidak berubah | Anda tidak lagi dikenai biaya instans segera setelah statusnya berubah menjadi `stopping`. Setiap kali sebuah instans bertransisi dari `stopped` ke `running`, kami memulai periode tagihan instans baru, yang menagih minimum satu menit setiap kali Anda memulai instans Anda. | Anda dikenai biaya saat instans berada dalam status `stopping`, tetapi tidak lagi dikenai biaya saat instans ada dalam status `stopped`. Setiap kali sebuah instans bertransisi dari `stopped` ke `running`, kami memulai periode tagihan instans baru, yang menagih minimum satu menit setiap kali Anda memulai instans Anda. | Anda berhenti menimbulkan biaya untuk suatu instans segera setelah statusnya berubah menjadi `shutting-down` |

Perintah shutdown sistem operasi selalu mengakhiri instance dengan volume root penyimpanan instance. Anda dapat mengontrol apakah perintah shutdown sistem operasi menghentikan atau menghentikan instance dengan volume root EBS. Lihat informasi yang lebih lengkap di Ubah perilaku shutdown yang diprakarsai instans.

# Menghentikan dan memulai instans Amazon EC2

Anda dapat menghentikan dan memulai instans jika memiliki volume Amazon EBS sebagai volume root. Saat Anda menghentikan sebuah instance, instans akan dimatikan. Ketika Anda memulai sebuah instance, biasanya dimigrasikan ke komputer host baru yang mendasarinya dan ditetapkan alamat IPv4 publik baru.

Penghentian instans dapat dimulai oleh pengguna (di mana Anda menghentikan instans secara manual) atau diprakarsai oleh AWS (sebagai respons terhadap peristiwa penghentian terjadwal saat AWS mendeteksi kegagalan host yang mendasari instans yang tidak dapat diperbaiki).

Untuk penghentian yang diprakarsai pengguna, sebaiknya gunakan konsol Amazon EC2, CLI, atau API alih-alih menjalankan perintah penghentian sistem operasi dari instans Anda. Saat menggunakan Amazon EC2, jika instans tidak mati secara bersih dalam beberapa menit, Amazon EC2 akan melakukan hard shutdown. Selanjutnya, AWS CloudTrail membuat catatan API kapan instance Anda dihentikan.

Topik ini menjelaskan cara melakukan penghentian yang diprakarsai pengguna. Untuk informasi tentang pemberhentian yang dilakukan oleh AWS, lihatKelola instans Amazon EC2 yang dijadwalkan untuk dihentikan atau dihentikan.

Saat Anda menghentikan sebuah instans, instans tersebut tidak dihapus. Jika Anda memutuskan bahwa Anda tidak lagi membutuhkan sebuah instans, Anda dapat mengakhirinya. Untuk informasi selengkapnya, lihat Menghentikan instans Amazon EC2. Jika Anda ingin menghibernasi instans untuk menyimpan konten dari memori instans (RAM), lihat Hibernasi instans Amazon EC2 Anda. Untuk perbedaan antara tindakan siklus hidup instans, lihat Perbedaan antara status instance.

Daftar Isi


- Cara kerja stop and start instance EC2
- Metode untuk menghentikan instance
- Menghentikan dan memulai instans Anda secara manual
- Menghentikan dan memulai instans Anda secara otomatis
- Temukan semua instans yang berjalan dan berhenti
- Temukan waktu peluncuran awal dan terbaru
- Aktifkan perlindungan berhenti untuk instans EC2 Anda

# Cara kerja stop and start instance EC2

Saat Anda menghentikan instans Amazon EC2, perubahan didaftarkan pada tingkat sistem operasi (OS) instans, beberapa sumber daya hilang, dan beberapa sumber daya tetap ada. Saat Anda memulai sebuah instans, perubahan terdaftar di tingkat instans.

Topik


- Apa yang terjadi jika Anda menghentikan sebuah instans
- Apa yang terjadi jika Anda memulai sebuah instans
- Uji respons aplikasi untuk berhenti dan mulai
- Biaya yang terkait dengan penghentian dan mulai instans


## Apa yang terjadi jika Anda menghentikan sebuah instans

Berikut ini menjelaskan apa yang biasanya terjadi ketika Anda menghentikan instance menggunakan metode berhenti default. Perhatikan bahwa beberapa aspek mungkin bervariasi tergantung pada metode berhenti yang Anda gunakan.

Perubahan terdaftar di tingkat OS

- Permintaan API akan mengirimkan peristiwa penekanan tombol kepada tamu.
- Berbagai layanan sistem dihentikan sebagai akibat dari peristiwa penekanan tombol. Pematian OS yang anggun dipicu oleh peristiwa penekanan tombol shutdown ACPI dari hypervisor.
- Pematian ACPI dimulai.
- Instans dimatikan ketika proses shutdown OS yang anggun keluar. Tidak ada waktu pematian OS yang dapat dikonfigurasi.
- Jika OS instans tidak dimatikan dengan bersih dalam beberapa menit, pematian keras dilakukan.
- Instans tersebut berhenti berjalan.
- Status instance berubah menjadi `stopping` dan kemudian`stopped`.
- [Penskalaan Otomatis] Jika instans Anda berada dalam grup Penskalaan Otomatis, ketika instans berada dalam status Amazon EC2 selain`running`, atau jika statusnya untuk pemeriksaan status menjadi`impaired`, Amazon EC2 Auto Scaling menganggap instans tidak sehat dan menggantikannya. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan untuk instans dalam grup Penskalaan Otomatis di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
- [Instans Windows] Saat Anda menghentikan dan memulai instance Windows, agen peluncuran melakukan tugas pada instans, seperti mengubah huruf drive untuk volume Amazon EBS

yang terlampir. Untuk informasi selengkapnya tentang default ini dan bagaimana Anda dapat mengubahnya, lihatthe section called “EC2Launch v2”.

Sumber daya hilang

- Data disimpan pada RAM.
- Data disimpan di volume penyimpanan instans.
- Alamat IPv4 publik yang secara otomatis ditetapkan Amazon EC2 ke instans saat diluncurkan atau dimulai. Untuk mempertahankan alamat IPv4 publik yang tidak pernah berubah, Anda dapat mengaitkan alamat IP Elastis dengan instans Anda.

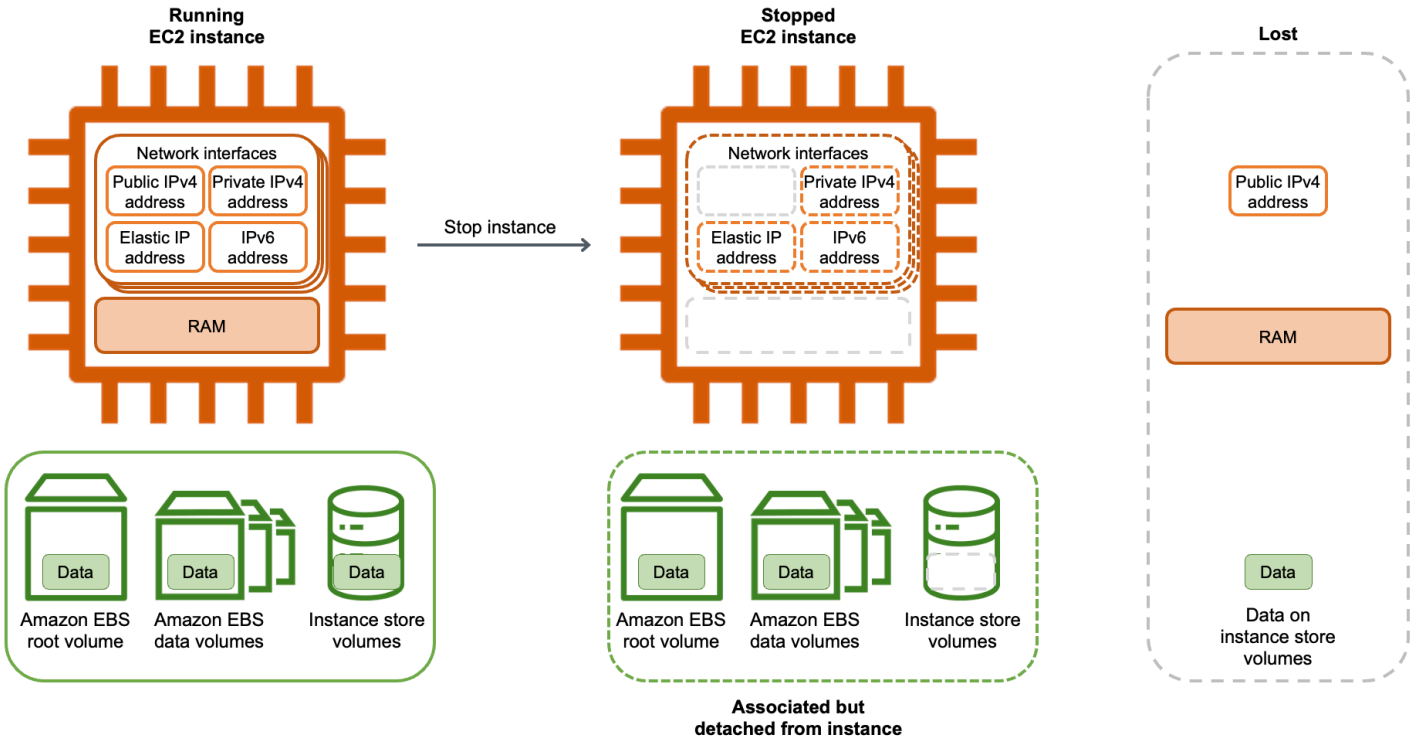
Sumber daya yang bertahan

- Semua volume root dan data Amazon EBS terlampir.
- Data disimpan di volume Amazon EBS.
- Setiap antarmuka jaringan terlampir.

  Antarmuka jaringan mencakup sumber daya berikut, yang juga bertahan:

  - Alamat IPv4 privat.
  - Alamat IPv6.
  - Alamat IP Elastis terkait dengan instans. Perhatikan bahwa ketika instans dihentikan, Anda akan dikenakan biaya untuk alamat IP Elastis terkait.

Diagram berikut menggambarkan apa yang tetap ada dan apa yang hilang ketika instance EC2 dihentikan. Diagram dibagi menjadi tiga bagian: bagian pertama, berlabel Running EC2 instance, menunjukkan instance dalam `running` status dengan sumber dayanya. Bagian kedua, berlabel Inst ance EC2 yang Dihentikan, menunjukkan instance dalam `stopped` status dengan sumber daya yang bertahan. Bagian ketiga, berlabel Hil ang, menunjukkan sumber daya yang hilang saat instance dihentikan.




Untuk informasi tentang apa yang terjadi saat Anda menghentikan instans Mac, lihatMenghentikan atau menghentikan instans Mac Amazon EC2 Anda.

Apa yang terjadi jika Anda memulai sebuah instans

- Dalam kebanyakan kasus, instans dimigrasikan ke komputer host dasar yang baru (meskipun dalam beberapa kasus instans tetap di host saat ini, seperti ketika sebuah instans dialokasikan ke host dalam konfigurasi Host Khusus).
- Volume EBS terkait dan antarmuka jaringan dilampirkan kembali ke instance.
- Amazon EC2 menetapkan alamat IPv4 publik baru ke instans jika instans dikonfigurasi untuk menerima alamat IPv4 publik, kecuali instans memiliki antarmuka jaringan sekunder atau alamat IPv4 pribadi sekunder yang terkait dengan alamat IP Elastic.
- Jika Anda menghentikan satu instans dalam grup penempatan dan kemudian memulainya lagi, ini masih berjalan dalam grup penempatan. Namun demikian, momen mulai gagal jika tidak cukup kapasitas untuk instans. Jika Anda menerima kesalahan kapasitas saat memulai instance di grup penempatan yang sudah menjalankan instans, hentikan semua instans dalam grup penempatan dan mulai semuanya lagi. Memulai instans dapat memigrasikannya ke perangkat keras yang memiliki kapasitas untuk semua proses yang diminta.

Uji respons aplikasi untuk berhenti dan mulai

Anda dapat menggunakannya AWS Fault Injection Service untuk menguji bagaimana aplikasi Anda merespons saat instans dihentikan dan dimulai. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS Fault Injection Service.

Biaya yang terkait dengan penghentian dan mulai instans

Biaya berikut dikaitkan dengan menghentikan dan memulai sebuah instans.

Mengh entikan — Segera setelah status instans berubah menjadi `shutting-down` atau`terminated`, biaya tidak lagi dikenakan untuk instans tersebut. Anda tidak dikenakan biaya untuk penggunaan atau biaya transfer data untuk instans yang dihentikan. Biaya dikenakan untuk menyimpan volume penyimpanan Amazon EBS.

Mulai — Setiap kali Anda memulai instans yang dihentikan, Anda akan dikenakan biaya penggunaan minimal satu menit. Setelah satu menit, Anda dikenai biaya hanya untuk detik yang digunakan. Misalnya, jika Anda menjalankan instans selama 20 detik, lalu menghentikannya, Anda akan dikenai biaya satu menit penggunaan. Jika Anda menjalankan instans selama 3 menit 40 detik, Anda dikenai biaya 3 menit dan 40 detik penggunaan.

## Metode untuk menghentikan instance

Ada empat cara untuk melakukan penghentian yang diprakarsai pengguna: berhenti default, berhenti dengan lewati penghentian OS, paksa berhenti, dan paksa berhenti dengan skip OS shutdown. Tabel berikut membandingkan perbedaan utama antara metode berhenti:

| Metode berhenti | Tujuan utama | Kasus penggunaan | Perintah CLI |
|---|---|---|---|
| Berhenti default | Penghentian instans normal dengan upaya shutdown OS yang anggun. | Contoh tipikal berhenti. | `aws ec2 stop-instances \ --instance-id i-1234567890abcdef0` |
| Berhenti dengan lewati | Melewati shutdown OS yang anggun saat menghentikan instance. | Saat melewati penghentian OS yang anggun diperlukan. | `aws ec2 stop-instances \ --instance-id i-1234567` |

| Metode berhenti | Tujuan utama | Kasus penggunaan | Perintah CLI |
| --- | --- | --- | --- |
| penutupa OS | | | `890abcdef0 \ --skip-os-shutdown` |
| Berhenti paksa | Menangani contoh macet. Mencoba berhenti default terlebih dahulu; jika instance gagal berhenti, maka secara paksa menghentikan instance. | Ketika instance macet dalam `stopping` keadaan. | `aws ec2 stop-instances \ --instance-id i-1234567890abcdef0 \ --force` |
| Berhenti paksa dengan lewati penutupa OS | Paksa berhenti dan melewati shutdown OS yang anggun saat menghentikan instance. | Ketika paksa berhenti dan melewati shutdown OS yang anggun diperlukan. | `aws ec2 stop-instances \ --instance-id i-1234567890abcdef0 \ --force \ --skip-os-shutdown` |

Untuk petunjuk tentang cara menggunakan setiap metode, lihat berikut ini:

- Hentikan instance dengan shutdown OS yang anggun
- Hentikan instance dan lewati shutdown OS yang anggun
- Memaksa menghentikan instance

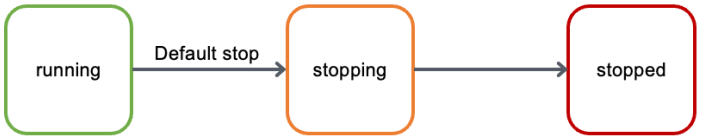
Metode hentikan:

- Berhenti default
- Berhenti dengan lewati penutupan OS
- Berhenti paksa
- Berhenti paksa dengan lewati penutupan OS

Bagian berikut memberikan informasi lebih rinci tentang empat metode penghentian yang berbeda yang diprakarsai pengguna.

### Berhenti default

Metode berhenti default adalah cara standar untuk menghentikan instance. Ketika Anda mengeluarkan StopInstances perintah, instance bertransisi dari `running` status, `kestopping`, dan akhirnya `kestopped`, seperti yang diilustrasikan oleh diagram berikut:




Tujuan: Shutdown instans  normal dengan upaya shutdown OS yang anggun.

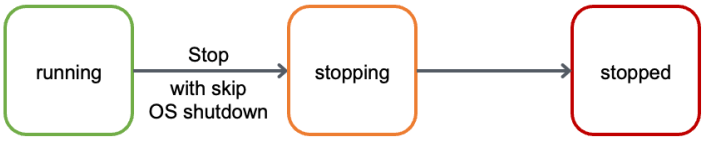
Dampak data: Mempertahankan data pada volume root EBS dan volume data. Kehilangan data pada volume penyimpanan instance.

Kapan menggunakan: Upaya pember  hentian pertama untuk pemberhentian biasa.

> **Note**
>
> Jika Anda sudah mencoba berhenti dengan skip OS shutdown, upaya berhenti default berikutnya selama sesi transisi status yang sama tidak akan melakukan shutdown OS yang anggun. Melewati shutdown OS yang anggun tidak dapat diubah untuk sesi instance saat ini.

### Berhenti dengan lewati penutupan OS

Ketika melewati shutdown OS yang anggun diperlukan, metode stop with skip OS shutdown dapat digunakan untuk menghentikan instance dan melewati shutdown OS yang anggun, seperti yang diilustrasikan oleh diagram berikut:



> **Warning**
>
> Melewati shutdown OS yang anggun dapat mengakibatkan hilangnya data atau kerusakan (misalnya, konten memori tidak dibuang ke disk atau hilangnya iOS dalam penerbangan) atau melewatkan skrip shutdown.

Tujuan: Lewati shutdown OS yang anggun saat menghentikan instance.

Dampak data: Dapat mengakibatkan hilangnya data atau korupsi. Isi memori mungkin tidak dibuang ke disk dan iOS dalam penerbangan mungkin hilang. Mungkin melewatkan skrip shutdown.

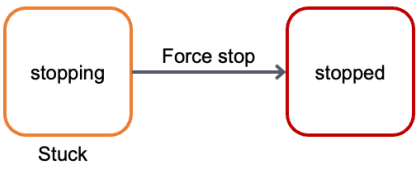
Kapan menggunakan: Saat melewati penghentian OS yang anggun diperlukan. Jika digunakan saat penghentian default dengan shutdown OS yang anggun sedang berlangsung, shutdown OS yang anggun akan dilewati.

> **Note**
>
> Melewati shutdown OS yang anggun tidak dapat diubah untuk sesi transisi status instance saat ini. Upaya berhenti default berikutnya selama sesi ini tidak akan mencoba shutdown OS yang anggun.

Berhenti paksa

Metode force stop digunakan untuk menangani instance yang macet dalam `stopping` status. Instans biasanya macet karena masalah perangkat keras yang mendasarinya (ditunjukkan oleh pemeriksaan status sistem yang gagal).

Metode force stop pertama kali mencoba berhenti default. Jika instance tetap macet dalam `stopping` status, `force` parameter secara paksa mematikan instance dan mentransisi instance ke `stopped` status, seperti yang ditunjukkan oleh diagram berikut:



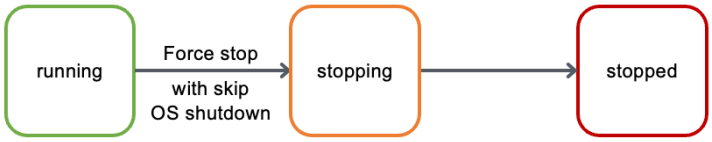

Tujuan: Men angani instans yang terjebak di `stopping` negara bagian. Coba berhenti default terlebih dahulu. Jika instance gagal berhenti, maka secara paksa mematikan instance.

Dampak data: Mencoba penghentian default terlebih dahulu, tetapi jika paksa berhenti dilakukan, maka dapat menyebabkan kehilangan atau kerusakan data. Dalam kasus yang jarang terjadi, menghasilkan penulisan pasca-penghentian ke volume EBS atau sumber daya bersama lainnya.

Kapan menggunakan: Up  aya berhenti kedua saat instance tetap macet setelah penghentian default. Untuk informasi selengkapnya, lihat Memecahkan masalah penghentian instans Amazon EC2.

Berhenti paksa dengan lewati penutupan OS

Ketika penghentian paksa dan melewati shutdown OS yang anggun diperlukan, metode force stop dengan skip OS shutdown dapat digunakan untuk membawa instance ke `stopped` status, seperti yang diilustrasikan dalam diagram berikut:




Tujuan:  Menggabungkan force stop dengan melewati shutdown OS yang anggun saat menghentikan instance.

Dampak data:  Melewati penutupan OS dapat mengakibatkan hilangnya atau korupsi data. Isi memori mungkin tidak dibuang ke disk dan iOS dalam penerbangan mungkin hilang. Mungkin melewatkan skrip shutdown. Jika force stop berlanjut, maka dapat menyebabkan hilangnya data tambahan atau korupsi. Dalam kasus yang jarang terjadi, menghasilkan penulisan pasca-berhenti ke volume EBS atau sumber daya bersama lainnya.

Kapan menggunakan:  Ketika Anda ingin memastikan bahwa instance Anda akan berhenti dan Anda ingin melewati shutdown OS yang anggun. Jika digunakan saat penghentian default dengan shutdown OS yang anggun sedang berlangsung, shutdown OS yang anggun akan dilewati.

## Menghentikan dan memulai instans Anda secara manual

Anda dapat menghentikan dan memulai instans Amazon Anda ( EBS-backed instans dengan volume root EBS). Anda tidak dapat menghentikan dan memulai instance dengan volume root penyimpanan instans.

Saat menggunakan metode default untuk menghentikan instance, shutdown sistem operasi (OS) yang anggun dicoba. Anda dapat melewati shutdown OS yang anggun; Namun, ini mungkin membahayakan integritas data.

> **Warning**
>
> Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Sebelum menghentikan instans, verifikasi bahwa Anda telah menyalin data apa pun yang Anda butuhkan dari volume penyimpanan instans ke penyimpanan persisten, seperti Amazon EBS atau Amazon S3.

[Instans Linux] Menggunakan halt perintah OS dari instance tidak memulai shutdown. Jika Anda menggunakan perintah halt, instans tidak berakhir; tetapi, instans akan menempatkan CPU ke `HLT`, yang menangguhkan operasi CPU. Instans tetap berjalan.

Anda dapat memulai shutdown menggunakan OS shutdown atau poweroff perintah. Saat Anda menggunakan perintah OS, instans berhenti secara default. Anda dapat mengubah perilaku ini. Untuk informasi selengkapnya, lihat Ubah perilaku shutdown yang diprakarsai instans.

> **Note**
>
> Jika Anda menghentikan EBS-backed instance Amazon dan tampaknya “macet” di `stopping` negara bagian, Anda dapat menghentikannya secara paksa. Untuk informasi selengkapnya, lihat Memecahkan masalah penghentian instans Amazon EC2.

Metode hentikan dan mulai

- Hentikan instance dengan shutdown OS yang anggun
- Hentikan instance dan lewati shutdown OS yang anggun
- Memulai instans

### Hentikan instance dengan shutdown OS yang anggun

Anda dapat menghentikan instance menggunakan metode stop default, yang mencakup upaya shutdown OS yang anggun. Untuk informasi selengkapnya, lihat Berhenti default.

Console

Untuk menghentikan instance menggunakan metode stop default

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi kiri, pilih  Instans, lalu pilih instans.
3. Pilih Status instans, Hentikan instans. Jika opsi ini dinonaktifkan, instans sudah dihentikan atau volume root adalah volume penyimpanan instans.
4. Ketika diminta konfirmasi, pilih Berhenti. Hal ini dapat memerlukan waktu beberapa menit sampai instans berhenti.

AWS CLI

Untuk menghentikan instance menggunakan metode stop default

Gunakan perintah stop-instance.

```
aws ec2 stop-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk menghentikan instance menggunakan metode stop default

Gunakan Stop-EC2Instance cmdlet

```
Stop-EC2Instance -InstanceId i-1234567890abcdef0
```

Hentikan instance dan lewati shutdown OS yang anggun

Anda dapat melewati shutdown OS yang anggun saat menghentikan instance. Untuk informasi selengkapnya, lihat Berhenti dengan lewati penutupan OS.

> **Warning**
>
> Melewati shutdown OS yang anggun dapat mengakibatkan hilangnya data atau kerusakan (misalnya, konten memori tidak dibuang ke disk atau hilangnya iOS dalam penerbangan) atau melewatkan skrip shutdown.

Console

Untuk menghentikan instance dan melewati shutdown OS yang anggun

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans, lalu pilih instans.
3. Pilih Keadaan nyata, Hentikan contoh.
4. Di bawah Lewati penutupan OS, pilih kotak centang Lewati penutupan OS. Jika Anda tidak melihat opsi ini di konsol, opsi ini belum tersedia di konsol di Wilayah saat ini. Namun, Anda dapat mengakses fitur ini menggunakan SDK AWS CLI atau, atau mencoba Wilayah lain di konsol.
5. Pilih Berhenti.

AWS CLI

Untuk menghentikan instance dan melewati shutdown OS yang anggun

Gunakan perintah stop-instance dengan. `--skip-os-shutdown`

```
aws ec2 stop-instances \
    --instance-ids i-1234567890abcdef0 \
    --skip-os-shutdown
```

PowerShell

Untuk menghentikan instance dan melewati shutdown OS yang anggun

Gunakan Stop-EC2Instance cmdlet dengan. `-SkipOsShutdown $true`

```
Stop-EC2Instance `
    -InstanceId i-1234567890abcdef0 `
    -SkipOsShutdown $true
```

Memulai instans

Anda dapat memulai instance yang dihentikan.

Console

Untuk memulai sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instance, dan pilih Status instans, Mulai instance.

Hal ini diperlukan waktu beberapa menit hingga instans memasuki status `running`.

AWS CLI

Untuk memulai sebuah instance

Gunakan perintah start-instances.

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk memulai sebuah instance

Gunakan Start-EC2Instance cmdlet.

```
Start-EC2Instance -InstanceId i-1234567890abcdef0
```

## Menghentikan dan memulai instans Anda secara otomatis

Anda dapat mengotomatisasi penghentian dan pemulaian instans dengan layanan berikut:

Penjadwal Instance aktif AWS

Anda dapat menggunakan Instance Scheduler aktif AWS untuk mengotomatiskan memulai dan menghentikan instans EC2. Untuk informasi selengkapnya, lihat Bagaimana cara menggunakan Penjadwal Instance CloudFormation untuk menjadwalkan instans EC2? Perhatikan bahwa biaya tambahan berlaku.

AWS Lambda dan EventBridge aturan Amazon

Anda dapat menggunakan Lambda dan EventBridge aturan untuk menghentikan dan memulai instance sesuai jadwal. Untuk informasi selengkapnya, lihat Bagaimana cara menggunakan Lambda untuk menghentikan dan memulai instans Amazon EC2 secara berkala?

Amazon EC2 Auto Scaling

Guna memastikan Anda memiliki jumlah instans Amazon EC2 yang tepat untuk menangani beban untuk aplikasi, buat grup Auto Scaling. Amazon EC2 Auto Scaling memastikan bahwa aplikasi Anda selalu memiliki kapasitas yang tepat untuk menangani permintaan lalu lintas, dan menghemat biaya dengan meluncurkan instans hanya saat dibutuhkan. Perhatikan bahwa

Amazon EC2 Auto Scaling mengakhiri, bukan menghentikan, instans yang tidak dibutuhkan. Untuk menyiapkan grup Auto Scaling, lihat Memulai Amazon EC2 Auto Scaling.

## Temukan semua instans yang berjalan dan berhenti

Anda dapat menemukan semua instans yang sedang berjalan dan dihentikan di semua instans Wilayah AWS di satu halaman menggunakan Amazon EC2 Global View. Kemampuan ini sangat berguna untuk mengambil inventaris dan menemukan instans yang terlupakan. Untuk informasi tentang cara menggunakan Tampilan Global, lihat Melihat sumber daya di seluruh Wilayah menggunakan AWS Tampilan Global.

Atau, Anda dapat menjalankan perintah atau cmdlet di setiap Wilayah tempat Anda memiliki instance.

AWS CLI

Untuk mendapatkan jumlah instans EC2 di suatu Wilayah

Gunakan perintah describe-instance berikut untuk menghitung instance di Wilayah saat ini. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance.

```
aws ec2 describe-instances \
    --region us-east-2 \
    --query "length(Reservations[].Instances[])"
```

Berikut ini adalah output contoh.

```
27
```

Untuk mendapatkan info ringkasan tentang instans EC2 Anda di Wilayah

Gunakan perintah describe-instance berikut. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance.

```
aws ec2 describe-instances \
    --region us-east-2 \
    --query "Reservations[].Instances[].[InstanceId,InstanceType,PrivateIpAddress]"
 \
    --output table
```

Berikut ini adalah output contoh.

```
-------------------------------------------------------
|                  DescribeInstances                  |
+---------------------+--------------+-----------------+
|  i-0e3e777f4362f1bf7|  t2.micro    |  10.0.12.9      |
|  i-09453945dcf1529e9|  t2.micro    |  10.0.143.213   |
|  i-08fd74f3f1595fdbd|  m7i.4xlarge |  10.0.1.103     |
+---------------------+--------------+-----------------+
```

PowerShell

Untuk mendapatkan jumlah instans EC2 di suatu Wilayah

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -Region us-east-2).Instances.Length
```

Berikut ini adalah output contoh.

```
27
```

Untuk mendapatkan info ringkasan tentang instans EC2 Anda di Wilayah

Gunakan Get-EC2Instance cmdlet. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance.

```
(Get-EC2Instance).Instances | Select InstanceId, InstanceType, PrivateIpAddress
```

Berikut ini adalah output contoh.

```
InstanceId          InstanceType PrivateIpAddress
----------          ------------ ----------------
i-0e3e777f4362f1bf7 t2.micro     10.0.12.9
i-09453945dcf1529e9 t2.micro     10.0.143.213
i-08fd74f3f1595fdbd m7i.4xlarge  10.0.1.103
```

## Temukan waktu peluncuran awal dan terbaru

Saat Anda mendeskripsikan sebuah instance, waktu peluncuran untuk instance tersebut adalah waktu peluncuran terbarunya. Setelah Anda menghentikan dan memulai sebuah instance, waktu

peluncuran mencerminkan waktu mulai instans baru. Untuk menemukan waktu peluncuran awal untuk sebuah instance, bahkan setelah menghentikan dan memulainya, lihat waktu di mana antarmuka jaringan utama dilampirkan ke instance.

Console

Untuk menemukan waktu peluncuran terbaru

Pilih instans dan temukan Waktu peluncuran di bawah Detail instans pada tab Detail.

Untuk menemukan waktu peluncuran awal

Pilih instance dan temukan antarmuka jaringan utama (indeks perangkat adalah 0) di bawah Antarmuka jaringan pada tab Jaringan.

AWS CLI

Untuk menemukan waktu peluncuran awal dan terbaru

Gunakan perintah describe-instance berikut untuk menampilkan waktu peluncuran awal dan waktu peluncuran terbaru untuk instance yang ditentukan.

```
aws ec2 describe-instances \
    --instance-id i-1234567890abcdef0 \
    --query 'Reservations[].Instances[].
{InstanceID:InstanceId,InitialLaunch:NetworkInterfaces[0].Attachment.AttachTime,LastLaunch:L
```

Berikut ini adalah output contoh.

```
[
    {
        "InstanceID": "i-1234567890abcdef0",
        "InitialLaunch": "2024-04-19T00:47:08+00:00",
        "LastLaunch": "2024-05-27T06:24:06+00:00"
    }
]
```

PowerShell

Untuk menemukan waktu peluncuran terbaru

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances.LaunchTime
```

Berikut ini adalah output contoh.

```
Monday, May 27, 2024 6:24:06 AM
```

Untuk menemukan waktu peluncuran awal

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -
InstanceId i-1234567890abcdef0).Instances.NetworkInterfaces.Attachment.AttachTime
```

Berikut ini adalah output contoh.

```
Friday, April 19, 2024 12:47:08 AM
```

## Aktifkan perlindungan berhenti untuk instans EC2 Anda

Untuk mencegah instans Anda berhenti secara tidak sengaja, Anda dapat mengaktifkan perlindungan penghentian untuk instans. Perlindungan penghentian juga melindungi instans Anda dari penghentian yang tidak disengaja.

A `DisableApiStop` tribut Amazon EC2 `ModifyInstanceAttribute` API mengontrol apakah instance dapat dihentikan dengan menggunakan konsol Amazon EC2, API AWS CLI, atau Amazon EC2. Anda dapat mengatur nilai atribut ini saat Anda meluncurkan instans, saat instans berjalan, atau saat instans berhenti.

Pertimbangan-pertimbangan

- Mengaktifkan perlindungan penghentian tidak menghindarkan Anda dari penghentian instans secara tidak sengaja dengan memulai pematian dari instans menggunakan perintah sistem operasi seperti shutdown atau poweroff.
- Mengaktifkan perlindungan penghentian tidak AWS mencegah penghentian instans ketika ada acara ter jadwal untuk menghentikan instans.
- Mengaktifkan perlindungan penghentian tidak mencegah Amazon EC2 Auto Scaling untuk menghentikan instans saat instans tidak sehat atau selama peristiwa penskalaan ke dalam. Anda

dapat mengontrol apakah grup Auto Scaling dapat mengakhiri instans tertentu saat meningkatkan skala dengan menggunakan perlindungan peningkatan skala instans.

- Perlindungan hentikan tidak hanya mencegah instans Anda dihentikan secara tidak sengaja, tetapi juga dari penghentian yang tidak disengaja saat menggunakan konsol AWS CLI,, atau API. Namun, itu tidak secara otomatis mengatur atribut `DisableApiTermination`. Perhatikan bahwa ketika `DisableApiStop` atribut disetel `kefalse`, pengaturan `DisableApiTermination` atribut menentukan apakah instance dapat dihentikan menggunakan konsol, AWS CLI, atau API. Untuk mengetahui informasi selengkapnya, lihat Menghentikan instans Amazon EC2.
- Anda tidak dapat mengaktifkan perlindungan penghentian untuk instans dengan volume root penyimpanan instans.
- Anda tidak dapat mengaktifkan perlindungan penghentian untuk Instans Spot.
- Amazon EC2 API mengikuti model konsistensi akhir saat Anda mengaktifkan atau menonaktifkan perlindungan penghentian. Ini berarti bahwa hasil dari menjalankan perintah untuk mengatur atribut perlindungan penghentian mungkin tidak langsung terlihat oleh semua perintah berikutnya yang Anda jalankan. Untuk informasi selengkapnya, lihat Konsistensi akhir di Panduan Pengembang Amazon EC2.

Hentikan tugas perlindungan

- Aktifkan perlindungan penghentian untuk instans saat peluncuran
- Aktifkan perlindungan penghentian untuk instans yang berjalan atau dihentikan
- Nonaktifkan perlindungan penghentian untuk instans yang berjalan atau berhenti

Aktifkan perlindungan penghentian untuk instans saat peluncuran

Anda dapat mengaktifkan perlindungan berhenti untuk instance saat meluncurkan instance.

Console

Untuk mengaktifkan perlindungan penghentian untuk sebuah instans saat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada dasbor, pilih Luncurkan instans.
3. Konfigurasikan instans Anda di wizard peluncuran instans baru.
4. Di wizard, aktifkan perlindungan penghentian dengan memilih Aktifkan untuk Perlindungan penghentian di bawah Detail lanjutan.

AWS CLI

Untuk mengaktifkan perlindungan penghentian untuk sebuah instans saat peluncuran

Gunakan perintah run-instance untuk meluncurkan instance. Tambahkan parameter berikut.

```
--disable-api-stop
```

PowerShell

Untuk mengaktifkan perlindungan penghentian untuk sebuah instans saat peluncuran

Gunakan New-EC2Instance cmdlet untuk meluncurkan instance. Tambahkan parameter berikut.

```
-DisableApiStop $true
```

Aktifkan perlindungan penghentian untuk instans yang berjalan atau dihentikan

Anda dapat mengaktifkan perlindungan berhenti untuk instance saat instans sedang berjalan atau dihentikan.

Console

Untuk mengaktifkan perlindungan penghentian untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans, lalu pilih Tindakan > Pengaturan instans > Ubah perlindungan penghentian.
4. Pilih kotak centang Aktifkan, lalu pilih S impan.

AWS CLI

Untuk mengaktifkan perlindungan penghentian untuk sebuah instance

Gunakan perintah modify- instance-atribut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --disable-api-stop
```

PowerShell

Untuk mengaktifkan perlindungan penghentian untuk sebuah instance

Gunakan Edit-EC2InstanceAttribute cmdlet.

```
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -DisableApiStop $true
```

Nonaktifkan perlindungan penghentian untuk instans yang berjalan atau berhenti

Anda dapat menonaktifkan proteksi penghentian untuk instans yang sedang berjalan atau berhenti menggunakan salah satu metode berikut.

Console

Untuk menonaktifkan perlindungan penghentian untuk instans yang berjalan atau dihentikan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans, lalu pilih Actions, instans settings, Change stop protection.
4. Kosongkan kotak centang Aktifkan, lalu pilih S impan.

AWS CLI

Untuk menonaktifkan perlindungan penghentian untuk instans yang berjalan atau dihentikan

Gunakan perintah modify- instance-atribut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --no-disable-api-stop
```

PowerShell

Untuk menonaktifkan perlindungan berhenti untuk sebuah instance

Gunakan Edit-EC2InstanceAttribute cmdlet.

```
Edit-EC2InstanceAttribute `
```

```
    -InstanceId i-1234567890abcdef0 `
    -DisableApiStop $false
```

## Hibernasi instans Amazon EC2 Anda

Saat Anda menghibernasi sebuah instans, Amazon EC2 memberi sinyal pada sistem operasi untuk melakukan hibernasi (suspend-to-disk). Hibernasi menyimpan konten dari memori instans (RAM) ke volume root Amazon Elastic Block Store (Amazon EBS). Amazon EC2 mempertahankan volume root EBS instans dan semua volume data EBS yang terlampir. Saat instans Anda dimulai:

- Volume root EBS dipulihkan ke status sebelumnya
- Isi RAM dimuat ulang
- Proses yang sebelumnya berjalan pada instans dilanjutkan
- Volume data terlampir sebelumnya akan dilampirkan kembali dan instans akan mempertahankan ID instansnya

Anda dapat menghibernasi instans hanya jika diaktifkan untuk hibernasi dan memenuhi prasyarat hibernasi.

Jika sebuah instans atau aplikasi membutuhkan waktu lama untuk melakukan bootstrap dan membangun jejak memori agar menjadi produktif sepenuhnya, Anda dapat menggunakan hibernasi untuk menghangatkan instans. Untuk menghangatkan instance, Anda:

1. Luncurkan dengan hibernasi diaktifkan.
2. Bawa ke status yang diinginkan.
3. Hibernasi sehingga siap dilanjutkan ke kondisi yang diinginkan kapan pun dibutuhkan.

Anda tidak dikenai biaya untuk penggunaan instans untuk instans hibernasi saat berada di status `stopped` atau untuk transfer data saat konten RAM ditransfer ke volume root EBS. Anda dikenai biaya untuk penyimpanan volume EBS apa pun, termasuk penyimpanan untuk konten RAM.

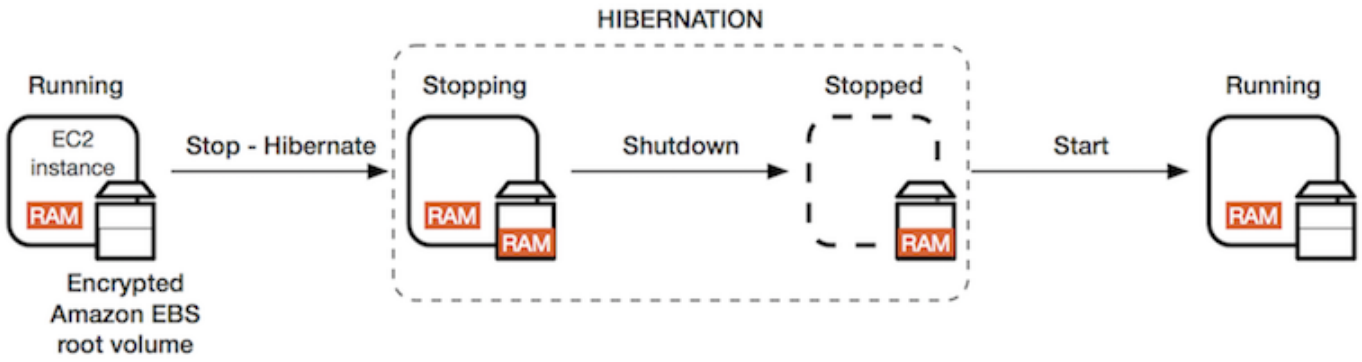
Jika Anda tidak lagi membutuhkan sebuah instans, Anda dapat mengakhirinya kapan saja, termasuk saat berada dalam status `stopped` (hibernasi). Untuk informasi selengkapnya, lihat Menghentikan instans Amazon EC2.

Daftar Isi

- Cara kerja hibernasi instans Amazon EC2
- Prasyarat untuk hibernasi instans EC2
- Konfigurasikan AMI Linux untuk mendukung hibernasi
- Aktifkan hibernasi untuk instans Amazon EC2
- Nonaktifkan KASLR pada instans (khusus Ubuntu)
- Hibernasi instans Amazon EC2
- Memulai instans Amazon EC2 yang hibernasi
- Memecahkan masalah hibernasi instans Amazon EC2


## Cara kerja hibernasi instans Amazon EC2

Diagram berikut menunjukkan ikhtisar dasar proses hibernasi untuk instans EC2.




Apa yang terjadi ketika Anda melakukan hibernasi sebuah instance

Saat Anda melakukan hibernasi instance, hal berikut terjadi:

- Contoh pindah ke `stopping` negara bagian. Amazon EC2 memberi sinyal pada sistem operasi untuk melakukan hibernasi (suspend-to-disk). Hibernasi membekukan semua proses, menyimpan konten RAM ke volume root EBS, dan kemudian melakukan shutdown secara teratur.
- Setelah penonaktifan selesai, instans berpindah ke status `stopped`.
- Setiap volume EBS tetap terlampir pada instans, dan data tetap ada, termasuk konten RAM yang disimpan.
- Setiap volume penyimpanan instans Amazon EC2 tetap melekat pada instans, tetapi data pada volume penyimpanan instans hilang.
- Dalam kebanyakan kasus, instance dipindahkan ke komputer host baru yang mendasarinya saat dimulai. Ini juga yang terjadi ketika Anda berhenti dan memulai sebuah instans.

- Saat instans dimulai, instans melakukan booting dan sistem operasi membaca konten RAM dari volume root EBS, sebelum membatalkan proses untuk melanjutkan statusnya.
- Instans mempertahankan alamat IPv4 privat-nya dan alamat IPv6. Saat instans dimulai, instans tersebut terus mempertahankan alamat IPv4 privatnya dan semua alamat IPv6.
- Amazon EC2 merilis alamat IPv4 publik. Saat instans dimulai, Amazon EC2 menetapkan alamat IPv4 publik baru ke instans.
- Instans mempertahankan alamat IP Elastis terkait Anda dikenai biaya untuk semua alamat IP Elastis yang terkait dengan instans hibernasi.

Untuk informasi tentang perbedaan hibernasi dari boot ulang, penghentian, dan pengakhiran, lihat Perbedaan antara status instance.

Batasan

- Ketika Anda menghibernasi suatu instans, data pada setiap volume penyimpanan instans akan hilang.
- (Instans Linux) Anda tidak dapat melakukan hibernasi instance Linux yang memiliki lebih dari 150 GiB RAM.
- (Instans Windows) Anda tidak dapat melakukan hibernasi instance Windows yang memiliki lebih dari 16 GiB RAM.
- Saat instance Anda hibernasi, Anda tidak dapat memodifikasinya. Ini berbeda dengan instance yang dihentikan yang tidak hibernasi, di mana Anda dapat memodifikasi atribut tertentu, seperti jenis atau ukuran instance.
- Jika Anda membuat snapshot atau AMI dari instans yang hibernasi atau mengaktifkan hibernasi, Anda mungkin tidak dapat terhubung ke instans baru yang diluncurkan dari AMI, atau dari AMI yang dibuat dari snapshot.
- (Instans Spot saja) Jika Amazon EC2 menghibernasi Instans Spot Anda, hanya Amazon EC2 yang dapat melanjutkan instans Anda. Jika Anda hibernasi instans Spot (hibernasi yang dimulai pengguna), Anda dapat melanjutkan instans Anda. Instans Spot hibernasi hanya dapat dilanjutkan jika kapasitas tersedia dan harga Spot kurang dari atau sama dengan harga maksimum yang Anda tentukan.
- Anda tidak dapat menghibernasi instans yang berada dalam grup Auto Scaling atau digunakan oleh Amazon ECS. Jika instans Anda berada dalam grup Auto Scaling dan Anda mencoba untuk hibernasi, layanan Amazon EC2 Auto Scaling menandai instans yang dihentikan sebagai tidak sehat, dan mungkin mengakhirinya serta meluncurkan instans pengganti. Untuk informasi

selengkapnya, lihat Pemeriksaan kesehatan untuk instans dalam grup Penskalaan Otom atis di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

- Anda tidak dapat melakukan hibernasi instance yang dikonfigurasi untuk boot dalam mode UEFI dengan Boot Aman UEFI diaktifkan.
- Jika Anda menghibernasi instans yang diluncurkan ke sebuah Reservasi Kapasitas, maka Reservasi Kapasitas tersebut tidak memastikan apakah instans yang dihibernasi dapat melanjutkan setelah Anda mencoba untuk memulainya.
- Anda tidak dapat menghibernasi instans yang menggunakan kernel di bawah 5.10 jika mode Federal Information Processing Standard (FIPS) diaktifkan.
- Kami tidak mendukung penyimpanan instans dalam mode hibernasi selama lebih dari 60 hari. Untuk mempertahankan instans lebih dari 60 hari, Anda harus memulai instans hibernasi, menghentikan instans, dan memulainya.
- Kami terus memperbarui platform kami dengan peningkatan dan tambalan keamanan, yang dapat bertentangan dengan instans hibernasi yang ada. Kami memberi tahu Anda tentang pembaruan penting yang memerlukan pemulaian untuk instans hibernasi sehingga kami dapat melakukan pematian atau boot ulang untuk menerapkan pemutakhiran dan patch keamanan yang diperlukan.

Pertimbangan untuk menghibernasi instans Spot

- Jika Anda menghibernasi Instans Spot, Anda hanya dapat memulai ulang jika kapasitas tersedia dan harga Spot kurang dari atau sama dengan harga maksimum yang Anda tentukan.
- Jika Amazon EC2 menghibernasi Instans Spot Anda:
  - Hanya Amazon EC2 yang dapat melanjutkan instans Anda.
  - Amazon EC2 melanjutkan Instans Spot hibernasi jika kapasitas tersedia dan harga Spot kurang dari atau sama dengan harga maksimum yang Anda tentukan.
  - Sebelum Amazon EC2 menghibernasi Instans Spot, Anda akan menerima pemberitahuan interupsi dua menit sebelum hibernasi dimulai.

  Untuk informasi selengkapnya, lihat Interupsi Instans Spot.

## Prasyarat untuk hibernasi instans EC2

Anda dapat mengaktifkan dukungan hibernasi untuk Inst On-Demand ans atau Instans Spot saat Anda meluncurkannya. Anda tidak dapat mengaktifkan hibernasi pada instans yang sudah ada,

apakah itu sedang berjalan atau dihentikan. Untuk informasi selengkapnya, lihat Aktifkan hibernasi instans.

Persyaratan untuk melakukan hibernasi instance

- Wilayah AWS
- AMI
- Keluarga instans
- Ukuran RAM instans
- Tipe volume root
- Ukuran volume root
- Enkripsi volume root
- Jenis Volume EBS
- Permintaan Instans Spot

Wilayah AWS

Anda dapat menggunakan hibernasi dengan instance di semua Wilayah AWS.

AMI

Anda harus menggunakan AMI HVM yang mendukung hibernasi. AMI berikut mendukung hibernasi:

AMI Linux

AMI untuk jenis instans Intel dan AMD

- AL2023 AMI dirilis 2023.09.20 atau yang lebih baru [1]
- AMI Amazon Linux 2 yang dirilis 29.08.2019 atau setelahnya
- AMI Amazon Linux 2018.03 yang dirilis 16.11.2018 atau setelahnya
- CentOS versi 8 AMI [2] (Konfigur asi tambahan diperlukan)
- Fedora versi 34 atau yang lebih baru AMI [2] (Konfigurasi tambahan diperlukan)
- Red Hat Enterprise Linux (RHEL) 9 AMI [2] (Diper ??? lukan konfigurasi tambahan)
- Red Hat Enterprise Linux (RHEL) 8 AMI [2] (Diper ??? lukan konfigurasi tambahan)
- Ubuntu 22.04.2 LTS (Jammy Jellyfish) AMI dirilis dengan nomor seri 20230303 atau yang lebih baru [3]

- Ubuntu 20.04 LTS (Focal Fossa) AMI dirilis dengan nomor seri 20210820 atau yang lebih baru [3]
- Ubuntu 18.04 LTS (Bionic Beaver) AMI dirilis dengan nomor seri 20190722.1 atau yang lebih baru [3]
- Ubuntu 16.04 LTS (Xenial Xerus) AMI [3] [4] ⇓ (Diperlukan konfigurasi ??? tambahan)

AMI untuk tipe instance Graviton

- AL2023 AMI (Arm 64-bit) dirilis 2024.07.01 atau yang lebih baru [1]
- Amazon Linux 2 AMI (64-bit Arm) dirilis 2024.06.20 atau lebih baru
- Ubuntu 22.04.2 LTS (64-bit Arm) (Jammy Jellyfish) AMI dirilis dengan nomor seri 20240701 atau yang lebih baru [3]
- Ubuntu 20.04 LTS (64-bit Arm) (Focal Fossa) AMI dirilis dengan nomor seri 20240701 atau yang lebih baru [3]

[1] Untuk AMI minimal AL2023, konfigurasi tambahan diperlukan.

[2] Untuk CentOS, Fedora, dan Red Hat Enterprise Linux, hibernasi hanya didukung pada Nitro-based instance.

[3] Kami merekomendasikan menonaktifkan KASLR pada instance dengan Ubuntu 22.04.2 LTS (Jammy Jellyfish), Ubuntu 20.04 LTS (Focal Fossa), Ubuntu 18.04 LTS (Bionic Beaver), dan Ubuntu 16.04 LTS (Xenial Xerus). Untuk informasi selengkapnya, lihat Nonaktifkan KASLR pada instans (khusus Ubuntu).

[4] Untuk AMI Ubuntu 16.04 LTS (Xenial Xerus), hibernasi tidak didukung pada jenis instans. `t3.nano` Tidak ada tambalan yang akan tersedia karena Ubuntu (Xenial Xerus) mengakhiri dukungan pada April 2021. Jika Anda ingin menggunakan tipe instans `t3.nano`, kami sarankan Anda untuk memutakhirkan ke Ubuntu 22.04.2 LTS (Jammy Jellyfish), Ubuntu 20.04 LTS (Focal Fossa) AMI, atau AMI Ubuntu 18.04 LTS (Bionic Beaver).

5 Dukungan untuk Ubuntu 18.04 LTS (Bionic Beaver) dan Ubuntu 16.04 LTS (Xenial Xerus) telah mencapai akhir masa pakai.

Untuk mengonfigurasi AMI Anda sendiri untuk mendukung hibernasi, lihat Konfigurasikan AMI Linux untuk mendukung hibernasi.

Dukungan untuk versi lain dari Ubuntu dan sistem operasi lain akan segera hadir.

AMI Windows

- AMI Windows Server 2022 yang dirilis 13.09.2023 atau setelahnya.
- AMI Windows Server 2019 yang dirilis 11.09.2019 atau setelahnya.
- AMI Windows Server 2016 yang dirilis 11.09.2019 atau setelahnya.
- AMI Windows Server 2012 R2 yang dirilis 11.09.2019 atau setelahnya
- AMI Windows Server 2012 yang dirilis 11.09.2019 atau setelahnya.

Keluarga instans

Anda harus menggunakan keluarga instance yang mendukung hibernasi. Namun, contoh bare metal tidak didukung.

- Tujuan umum: M3, M4, M5, M5a, M5ad, M5d, M6a, M6g, M6gd, M6i, M6id, M6idn, M6in, M7a, M7g, M7gd, M7i,, M8a, M8azN, M8g, M8gD, M8gD, M8gn, M8gn, M8gn, 8i,, M8in, M8idn, M8ib, M8iDB, M9g, M9gd, T2, T3 M7i-flex, T3a, T4g M8i-flex
- Komputasi dioptimalkan: C3, C4, C5, C5d, C6a, C6g, C6gD, C6gn, C6i, C6id, C6in, C7a, C7g, C7gD, C7gn, C7i,, C8a, C8g, C8g, C8gD, C8gn, C8i,, C8in, C8ib, C9g, C9gd C7i-flex C8i-flex
- Memori dioptimalkan: R3, R4, R5, R5a, R5ad, R5d, R6a, R6g, R6gd, R6idn, R6in, R7a, R7g, R7gd, R7i, R7iz, R8a, R8g, R8g, R8gd, R8gn, R8i, R8i, R8in, R8idn, R8ib, R8iDB, R9g, R9gd, X2Gd, X8AEDZ, X8i R8i-flex
- Penyimpanan dioptimalkan: I3, i3en, i4G, i7i, i7ie, i8G, i8GE, iM4GN, IS4gen

Console

Untuk mendapatkan jenis instance yang mendukung hibernasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Tipe Instans.
3. Tambahkan filter On-Demand Hibernation support = true.
4. (Opsional) Tambahkan filter ke cakupan lebih lanjut ke jenis instans tertentu yang menarik.

AWS CLI

Untuk mendapatkan jenis instance yang mendukung hibernasi

Gunakan perintah describe-instance-types. Perhatikan bahwa jenis instans yang tersedia bervariasi menurut Wilayah.

```
aws ec2 describe-instance-types \
    --filters Name=hibernation-supported,Values=true \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

PowerShell

Untuk mendapatkan jenis instance yang mendukung hibernasi

Gunakan Get-EC2InstanceType cmdlet. Perhatikan bahwa jenis instans yang tersedia bervariasi menurut Wilayah.

```
(Get-EC2InstanceType `
    -Filter @{Name="hibernation-supported"; Values="true"}).InstanceType | Sort-
Object
```

Ukuran RAM instans

Instans Linux — Harus kurang dari 150 GiB.

Instans Windows — Harus kurang dari atau sama dengan 16 GiB. Untuk hibernasi instance Windows T3 atau T3a, kami merekomendasikan setidaknya 1 GiB RAM.

Tipe volume root

Volume root harus berupa volume EBS, bukan volume penyimpanan instans.

Ukuran volume root

Volume root harus cukup besar untuk menyimpan konten RAM dan mengakomodasi penggunaan yang Anda harapkan, misalnya, OS atau aplikasi. Jika Anda mengaktifkan hibernasi, ruang dialokasikan pada volume root saat peluncuran untuk menyimpan RAM.

Enkripsi volume root

Volume root harus dienkripsi untuk melindungi konten sensitif yang ada dalam memori selama hibernasi. Anda dapat mengaktifkan hibernasi untuk instans melalui Amazon EC2 dengan menggunakan konsol, AWS CLI, atau API. Sebelum instance diluncurkan, Amazon EC2 memvalidasi

bahwa volume root dienkripsi. Karena volume root EBS dienkripsi, data RAM yang ditulis padanya selama hibernasi juga dienkripsi.

> ⚠ Hanya Amazon EC2 API yang memberlakukan enkripsi saat istirahat untuk hibernasi
>
> Amazon EC2 memvalidasi prasyarat hibernasi, termasuk enkripsi volume root, hanya untuk hibernasi yang Anda mulai melalui Amazon EC2. Ini termasuk konsol, AWS CLI, dan API. Pemeriksaan ini tidak berlaku jika Anda memulai hibernasi dari dalam sistem operasi tamu, seperti Linux atau Windows suspend-to-disk. Dalam hal ini, pastikan bahwa volume root EBS dienkripsi. Sistem operasi mempertahankan isi memori ke volume ini sebelum menghentikan instance. Di bawah model tanggung jawab bersama, Anda bertanggung jawab untuk mengenkripsi data saat diam untuk tindakan yang Anda lakukan di sistem operasi tamu. Untuk informasi selengkapnya, lihat Model tanggung jawab bersama dan Melindungi data saat diam di Pilar Keamanan - Ker AWS Well-Architected angka Kerja.

Gunakan salah satu dari tiga opsi berikut untuk memastikan bahwa volume root adalah volume EBS terenkripsi:

- Enkripsi EBS secara default — Anda dapat mengaktifkan enkripsi EBS secara default untuk memastikan bahwa semua volume EBS baru yang dibuat di AWS akun Anda dienkripsi. Dengan cara ini, Anda dapat mengaktifkan hibernasi untuk instans Anda tanpa menentukan maksud enkripsi pada peluncuran instans. Untuk informasi selengkapnya, lihat Mengaktifkan enkripsi secara default.
- Enkripsi “satu langkah” EBS — Anda dapat meluncurkan instans EBS-backed EC2 terenkripsi dari AMI yang tidak terenkripsi dan juga mengaktifkan hibernasi pada saat yang bersamaan. Untuk informasi selengkapnya, lihat Gunakan enkripsi dengan EBS-backed AMI.
- AMI terenkrips – Anda dapat mengaktifkan enkripsi EBS dengan menggunakan AMI terenkripsi untuk meluncurkan instans Anda. Jika AMI Anda tidak memiliki snapshot root terenkripsi, Anda dapat menyalinnya ke AMI baru dan meminta enkripsi. Untuk informasi selengkapnya, lihat Mengenkripsikan gambar yang tidak dienkripsi selama penyalinan dan Menyalin AMI.

### Jenis Volume EBS

Volume EBS harus menggunakan salah satu jenis volume EBS berikut:

- SSD Tujuan Umum (gp2 dan gp3)

- SSD IOPS yang Tersedia (`io1` dan `io2`)

Jika Anda memilih tipe volume SSD IOPS yang Tersedia, Anda harus menyediakan volume EBS dengan IOPS yang sesuai untuk mencapai performa yang optimal untuk hibernasi. Untuk informasi selengkapnya, lihat Jenis volume Amazon EBS di Panduan Pengguna Amazon EBS.

Permintaan Instans Spot

Untuk Instans Spot, persyaratan berikut berlaku:

- Tipe permintaan Instans Spot harus `persistent`.
- Anda tidak dapat menentukan grup peluncuran dalam permintaan Instans Spot.

## Konfigurasikan AMI Linux untuk mendukung hibernasi

AMI Linux berikut dapat mendukung hibernasi instans Amazon EC2, asalkan Anda menyelesaikan langkah-langkah konfigurasi tambahan yang dijelaskan di bagian ini.

Konfigurasi tambahan diperlukan untuk:

- AL2023 AMI minimal dirilis 2023.09.20 atau lebih baru
- AMI minimal Amazon Linux 2 yang dirilis 29.08.2019 atau setelahnya
- AMI Amazon Linux 2 yang dirilis 29.08.2019
- Amazon Linux yang dirilis sebelum 16.11.2018
- CentOS versi 8 atau setelahnya
- Fedora versi 34 atau setelahnya
- Red Hat Enterprise Linux versi 8 atau 9
- Ubuntu 20.04 LTS (Focal Fossa) dirilis sebelum nomor seri 20210820
- Ubuntu 18.04 (Bionic Beaver) dirilis dengan nomor seri 20190722.1
- Ubuntu 16.04 (Xenial Xerus)

Untuk AMI Linux dan Windows yang mendukung hibernasi dan tidak diperlukan konfigurasi tambahan, lihatAMI.

Untuk informasi selengkapnya, lihat Memperbarui perangkat lunak instans pada instans Amazon Linux 2 Anda.

AL2023 AMI minimal dirilis 2023.09.20 atau lebih baru

Untuk mengonfigurasi AMI minimal AL2023 yang dirilis 2023.09.20 atau lebih baru untuk mendukung hibernasi

1. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo dnf install ec2-hibinit-agent
```

2. Mulai ulang layanan.

```
[ec2-user ~]$ sudo systemctl start hibinit-agent
```

AMI minimal Amazon Linux 2 yang dirilis 29.08.2019 atau setelahnya

Untuk mengonfigurasi AMI minimal Amazon Linux 2 yang dirilis 29.08.2019 atau setelahnya untuk mendukung hibernasi

1. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

2. Mulai ulang layanan.

```
[ec2-user ~]$ sudo systemctl start hibinit-agent
```

AMI Amazon Linux 2 yang dirilis 29.08.2019

Untuk mengonfigurasi AMI Amazon Linux 2 yang dirilis sebelum 29.08.2019 untuk mendukung hibernasi

1. Perbarui kernel ke `4.14.138-114.102` atau lebih baru.

```
[ec2-user ~]$ sudo yum update kernel
```

2. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

3. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

4. Konfirmasikan bahwa versi kernel telah diperbarui ke `4.14.138-114.102` atau setelahnya.

```
[ec2-user ~]$ uname -a
```

5. Hentikan instans dan buat AMI. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.

Amazon Linux yang dirilis sebelum 16.11.2018

Untuk mengonfigurasi AMI Amazon Linux 2 yang dirilis sebelum 16.11.2018 untuk mendukung hibernasi

1. Perbarui kernel ke `4.14.77-70.59` atau lebih baru.

```
[ec2-user ~]$ sudo yum update kernel
```

2. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

3. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

4. Konfirmasikan bahwa versi kernel telah diperbarui ke `4.14.77-70.59` atau yang lebih tinggi.

```
[ec2-user ~]$ uname -a
```

5. Hentikan instans dan buat AMI. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.

CentOS versi 8 atau setelahnya

Untuk mengonfigurasi CentOS versi 8 atau AMI setelahnya untuk mendukung hibernasi

1. Perbarui kernel ke `4.18.0-305.7.1.el8_4.x86_64` atau lebih baru.

```
[ec2-user ~]$ sudo yum update kernel
```

2. Instal repositori Fedora Extra Packages for Enterprise Linux (EPEL).

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

3. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

4. Aktifkan agen hibernasi untuk memulai saat boot.

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

5. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

6. Konfirmasikan bahwa versi kernel telah diperbarui ke `4.18.0-305.7.1.el8_4.x86_64` atau setelahnya.

```
[ec2-user ~]$ uname -a
```

Fedora versi 34 atau setelahnya

Untuk mengkonfigurasi Fedora versi 34 atau AMI setelahnya untuk mendukung hibernasi

1. Perbarui kernel ke `5.12.10-300.fc34.x86_64` atau lebih baru.

```
[ec2-user ~]$ sudo yum update kernel
```

2. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo dnf install ec2-hibinit-agent
```

3. Aktifkan agen hibernasi untuk memulai saat boot.

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

4. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

5. Konfirmasikan bahwa versi kernel telah diperbarui ke `5.12.10-300.fc34.x86_64` atau setelahnya.

```
[ec2-user ~]$ uname -a
```

Red Hat Enterprise Linux versi 8 atau 9

Untuk mengonfigurasi Red Hat Enterprise Linux 8 atau 9 AMI untuk mendukung hibernasi

1. Perbarui kernel ke `4.18.0-305.7.1.el8_4.x86_64` atau lebih baru.

```
[ec2-user ~]$ sudo yum update kernel
```

2. Instal repositori Fedora Extra Packages for Enterprise Linux (EPEL).

   RHEL versi 8:

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

   RHEL versi 9:

```
[ec2-user ~]$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-9.noarch.rpm
```

3. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo yum install ec2-hibinit-agent
```

4. Aktifkan agen hibernasi untuk memulai saat boot.

```
[ec2-user ~]$ sudo systemctl enable hibinit-agent.service
```

5. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

6. Konfirmasikan bahwa versi kernel telah diperbarui ke `4.18.0-305.7.1.el8_4.x86_64` atau setelahnya.

```
[ec2-user ~]$ uname -a
```

Ubuntu 20.04 LTS (Focal Fossa) dirilis sebelum nomor seri 20210820

Untuk mengonfigurasi AMI Ubuntu 20.04 LTS (Focal Fossa) yang dirilis sebelum nomor seri 20210820 untuk mendukung hibernasi

1. Perbarui linux-aws-kernel ke `5.8.0-1038.40` atau setelahnya, dan grub2 ke `2.04-1ubuntu26.13` atau setelahnya.

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt dist-upgrade
```

2. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

3. Konfirmasikan bahwa versi kernel telah diperbarui ke `5.8.0-1038.40` atau setelahnya.

```
[ec2-user ~]$ uname -a
```

4. Konfirmasikan bahwa versi grub2 diperbarui ke `2.04-1ubuntu26.13` atau lebih baru.

```
[ec2-user ~]$ dpkg --list | grep grub2-common
```

Ubuntu 18.04 (Bionic Beaver) dirilis dengan nomor seri 20190722.1

Untuk mengonfigurasi AMI Ubuntu 18.04 LTS yang dirilis sebelum nomor seri 20190722.1 untuk mendukung hibernasi

1. Perbarui kernel ke `4.15.0-1044` atau lebih baru.

```
[ec2-user ~]$ sudo apt update
```

```
[ec2-user ~]$ sudo apt dist-upgrade
```

2. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo apt install ec2-hibinit-agent
```

3. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

4. Konfirmasikan bahwa versi kernel telah diperbarui ke `4.15.0-1044` atau setelahnya.

```
[ec2-user ~]$ uname -a
```

Ubuntu 16.04 (Xenial Xerus)

Untuk mengonfigurasi Ubuntu 16.04 LTS untuk mendukung hibernasi, Anda perlu menginstal paket kernel linux-aws-hwe versi 4.15.0-1058-aws atau setelahnya dan ec2-hibinit-agent.

> **Important**
>
> Paket kernel `linux-aws-hwe` didukung sepenuhnya oleh Canonical. Dukungan standar untuk Ubuntu 16.04 LTS berakhir pada April 2021, dan paket tidak lagi menerima pembaruan rutin. Namun, ini akan menerima pembaruan keamanan tambahan hingga dukungan Pemeliharaan Keamanan Diperpanjang berakhir pada 2024. Untuk informasi selengkapnya, lihat Hibernasi Amazon EC2 untuk Ubuntu 16.04 LTS sekarang tersedia di Blog Ubuntu Canonical.
>
> Kami menyarankan Anda untuk memutakhirkan ke AMI Ubuntu 20.04 LTS (Focal Fossa) atau AMI Ubuntu 18.04 LTS (Bionic Beaver).

Untuk mengonfigurasi AMI Ubuntu 16.04 LTS untuk mendukung hibernasi

1. Perbarui kernel ke `4.15.0-1058-aws` atau lebih baru.

```
[ec2-user ~]$ sudo apt update
[ec2-user ~]$ sudo apt install linux-aws-hwe
```

2. Instal paket `ec2-hibinit-agent` dari repositori.

```
[ec2-user ~]$ sudo apt install ec2-hibinit-agent
```

3. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

4. Konfirmasikan bahwa versi kernel telah diperbarui ke `4.15.0-1058-aws` atau setelahnya.

```
[ec2-user ~]$ uname -a
```

## Aktifkan hibernasi untuk instans Amazon EC2

Untuk menghibernasi instans, Anda harus terlebih dahulu mengaktifkannya untuk hibernasi saat meluncurkan instans.

> **Important**
>
> Anda tidak dapat mengaktifkan atau menonaktifkan hibernasi untuk sebuah instans setelah Anda meluncurkannya.

Tugas

- Aktifkan hibernasi untuk Instans On-Demand
- Aktifkan hibernasi untuk Instans Spot
- Untuk melihat apakah instans diaktifkan untuk hibernasi

### Aktifkan hibernasi untuk Instans On-Demand

Anda dapat mengaktifkan hibernasi untuk On-Demand Instans Anda.

Console

Untuk mengaktifkan hibernasi untuk Instance On-Demand

1. Ikuti prosedur untuk meluncurkan instans, tetapi jangan meluncurkan instans sampai Anda menyelesaikan langkah-langkah berikut untuk mengaktifkan hibernasi.
2. Untuk mengaktifkan hibernasi, konfigurasikan bidang berikut di wizard peluncuran instans:

a. Pada Aplikasi dan Gambar OS (Amazon Machine Image), pilih AMI yang mendukung hibernasi. Untuk informasi selengkapnya, lihat AMI.

b. Pada Tipe instans, pilih tipe instans yang didukung. Untuk informasi selengkapnya, lihat Keluarga instans.

c. Pada Konfigurasikan penyimpanan, pilih Lanjutan (di sebelah kanan), dan tentukan informasi berikut untuk volume root:

- Untuk Ukuran (GiB), masukkan ukuran volume root EBS. Volume harus cukup besar untuk menyimpan konten RAM dan mengakomodasi penggunaan yang Anda harapkan.
- Untuk Tipe volume, pilih tipe volume EBS yang didukung, SSD Tujuan Umum (`gp2` dan `gp3`) atau SSD IOPS yang Tersedia (`io1` dan `io2`).
- Untuk Terenkripsi, pilih Ya. Jika Anda mengaktifkan enkripsi secara default di Wil AWS ayah ini, Ya dipilih.
- Untuk Kunci KMS, pilih kunci enkripsi untuk volume. Jika Anda mengaktifkan enkripsi secara default di Wil AWS ayah ini, kunci enkripsi default dipilih.

Untuk informasi selengkapnya tentang prasyarat volume root, lihat Prasyarat untuk hibernasi instans EC2.

d. Perluas Detail lanjutan, dan untuk Perilaku Hentikan - Hibernasi, pilih Aktifkan.

3. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

AWS CLI

Untuk mengaktifkan hibernasi untuk Instance On-Demand

Gunakan perintah run-instances untuk meluncurkan instans. Tentukan parameter volume root EBS menggunakan parameter `--block-device-mappings file://mapping.json`, dan aktifkan hibernasi menggunakan parameter `--hibernation-options Configured=true`.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type m5.large \
    --block-device-mappings file://mapping.json \
```

```
    --hibernation-options Configured=true \
    --count 1 \
    --key-name MyKeyPair
```

Tentukan hal berikut dalam `mapping.json`.

```
[
    {
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "VolumeSize": 30,
            "VolumeType": "gp2",
            "Encrypted": true
        }
    }
]
```

Nilai untuk `DeviceName` harus cocok dengan nama perangkat root yang terkait dengan AMI. Untuk menemukan nama perangkat root, gunakan perintah describe-images.

```
aws ec2 describe-images --image-id ami-0abcdef1234567890
```

Jika Anda mengaktifkan enkripsi secara default di Wil AWS ayah ini, Anda dapat menghilangkannya`"Encrypted": true`.

PowerShell

Untuk mengaktifkan hibernasi untuk Instance On-Demand

Gunakan New-EC2Instance perintah untuk meluncurkan instance. Tentukan volume root EBS dengan menentukan pemetaan perangkat blok terlebih dahulu, lalu menambahkannya ke perintah menggunakan parameter `-BlockDeviceMappings`. Aktifkan hibernasi menggunakan parameter `-HibernationOptions_Configured $true`.

```
$ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
$ebs_encrypt.DeviceName = "/dev/xvda"
$ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
$ebs_encrypt.Ebs.VolumeSize = 30
$ebs_encrypt.Ebs.VolumeType = "gp2"
$ebs_encrypt.Ebs.Encrypted = $true
```

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType m5.large `
    -BlockDeviceMappings $ebs_encrypt `
    -HibernationOptions_Configured $true `
    -MinCount 1 `
    -MaxCount 1 `
    -KeyName MyKeyPair
```

Nilai untuk `DeviceName` harus cocok dengan nama perangkat root yang terkait dengan AMI. Untuk menemukan nama perangkat root, gunakan Get-EC2Image perintah.

```
Get-EC2Image -ImageId ami-0abcdef1234567890
```

Jika Anda mengaktifkan enkripsi secara default di AWS Wilayah ini, Anda dapat menghilangkan `Encrypted = $true` pemetaan perangkat blok.

### Aktifkan hibernasi untuk Instans Spot

Anda dapat mengaktifkan hibernasi untuk Instans Spot Anda. Untuk informasi selengkapnya tentang hibernasi instans Spot saat interupsi, lihat Interupsi Instans Spot.

#### Console

Untuk mengaktifkan hibernasi untuk Instans Spot

1. Ikuti prosedur untuk meminta Instans Spot menggunakan wizard peluncuran instans, tetapi jangan luncurkan instans sampai Anda menyelesaikan langkah-langkah berikut untuk mengaktifkan hibernasi.
2. Untuk mengaktifkan hibernasi, konfigurasikan bidang berikut di wizard peluncuran instans:
   a. Pada Aplikasi dan Gambar OS (Amazon Machine Image), pilih AMI yang mendukung hibernasi. Untuk informasi selengkapnya, lihat AMI.
   b. Pada Tipe instans, pilih tipe instans yang didukung. Untuk informasi selengkapnya, lihat Keluarga instans.
   c. Pada Konfigurasikan penyimpanan, pilih Lanjutan (di sebelah kanan), dan tentukan informasi berikut untuk volume root:

- Untuk Ukuran (GiB), masukkan ukuran volume root EBS. Volume harus cukup besar untuk menyimpan konten RAM dan mengakomodasi penggunaan yang Anda harapkan.
- Untuk Tipe volume, pilih tipe volume EBS yang didukung, SSD Tujuan Umum (`gp2` dan `gp3`) atau SSD IOPS yang Tersedia (`io1` dan `io2`).
- Untuk Terenkripsi, pilih Ya. Jika Anda mengaktifkan enkripsi secara default di Wil AWS ayah ini, Ya dipilih.
- Untuk Kunci KMS, pilih kunci enkripsi untuk volume. Jika Anda mengaktifkan enkripsi secara default di Wil AWS ayah ini, kunci enkripsi default dipilih.

Untuk informasi selengkapnya tentang prasyarat volume root, lihat Prasyarat untuk hibernasi instans EC2.

d. Perluas Detail lanjutan, dan, selain bidang untuk mengonfigurasi instans Spot, lakukan hal berikut:

i. Untuk Tipe permintaan, pilih Persisten.

ii. Untuk Perilaku interupsi, pilih Hibernasi. Atau, untuk perilaku Berhenti - Hibernasi, pilih Aktifkan. Kedua bidang mengaktifkan hibernasi pada Instans Spot Anda. Anda hanya perlu mengonfigurasi salah satunya.

3. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

AWS CLI

Untuk mengaktifkan hibernasi untuk Instans Spot

Gunakan perintah run-instances untuk meminta Instans Spot. Tentukan parameter volume root EBS menggunakan parameter `--block-device-mappings file://mapping.json`, dan aktifkan hibernasi menggunakan parameter `--hibernation-options Configured=true`. Tipe permintaan Spot (`SpotInstanceType`) harus `persistent`.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c4.xlarge \
    --block-device-mappings file://mapping.json \
```

```
    --hibernation-options Configured=true \
    --count 1 \
    --key-name MyKeyPair
    --instance-market-options
        {
           "MarketType":"spot",
           "SpotOptions":{
              "MaxPrice":"1",
              "SpotInstanceType":"persistent"
            }
        }
```

Tentukan parameter volume root EBS `mapping.json` sebagai berikut.

```
[
    {
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "VolumeSize": 30,
            "VolumeType": "gp2",
            "Encrypted": true
        }
    }
]
```

Nilai untuk `DeviceName` harus cocok dengan nama perangkat root yang terkait dengan AMI. Untuk menemukan nama perangkat root, gunakan perintah describe-images.

```
aws ec2 describe-images --image-id ami-0abcdef1234567890
```

Jika Anda mengaktifkan enkripsi secara default di Wil AWS ayah ini, Anda dapat menghilangkannya`"Encrypted": true`.

PowerShell

Untuk mengaktifkan hibernasi untuk Instans Spot

Gunakan New-EC2Instance perintah untuk meminta Instance Spot. Tentukan volume root EBS dengan menentukan pemetaan perangkat blok terlebih dahulu, lalu menambahkannya ke perintah menggunakan parameter `-BlockDeviceMappings`. Aktifkan hibernasi menggunakan parameter `-HibernationOptions_Configured $true`.

```
$ebs_encrypt = New-Object Amazon.EC2.Model.BlockDeviceMapping
$ebs_encrypt.DeviceName = "/dev/xvda"
$ebs_encrypt.Ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
$ebs_encrypt.Ebs.VolumeSize = 30
$ebs_encrypt.Ebs.VolumeType = "gp2"
$ebs_encrypt.Ebs.Encrypted = $true

New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType m5.large `
    -BlockDeviceMappings $ebs_encrypt `
    -HibernationOptions_Configured $true `
    -MinCount 1 `
    -MaxCount 1 `
    -KeyName MyKeyPair `
    -InstanceMarketOption @(
        MarketType = spot;
        SpotOptions @{
        MaxPrice = 1;
        SpotInstanceType = persistent}
    )
```

Nilai untuk `DeviceName` harus cocok dengan nama perangkat root yang terkait dengan AMI. Untuk menemukan nama perangkat root, gunakan Get-EC2Image perintah.

```
Get-EC2Image -ImageId ami-0abcdef1234567890
```

Jika Anda mengaktifkan enkripsi secara default di AWS Wilayah ini, Anda dapat menghilangkan `Encrypted = $true` pemetaan perangkat blok.

Untuk melihat apakah instans diaktifkan untuk hibernasi

Anda dapat memeriksa apakah instance diaktifkan untuk hibernasi.

Console

Untuk melihat apakah instans diaktifkan untuk hibernasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.

3. Pilih instance dan, pada tab Detail, di bagian Detail Instance, periksa Stop-hibernate perilaku. Enabled menunjukkan bahwa instans diaktifkan untuk hibernasi.

AWS CLI

Untuk melihat apakah instans diaktifkan untuk hibernasi

Gunakan perintah describe-instances dan tentukan parameter `--filters "Name=hibernation-options.configured,Values=true"` untuk memfilter instans yang diaktifkan untuk hibernasi.

```
aws ec2 describe-instances \
    --filters "Name=hibernation-options.configured,Values=true"
```

Bidang berikut di keluaran menunjukkan bahwa instans diaktifkan untuk hibernasi.

```
"HibernationOptions": {
    "Configured": true
}
```

PowerShell

Untuk melihat apakah instans diaktifkan untuk hibernasi

Gunakan Get-EC2Instance cmdlet dan filter instance yang diaktifkan untuk hibernasi.

```
(Get-EC2Instance `
    -Filter @{Name="hibernation-options.configured"; Values="true"}).Instances
```

## Nonaktifkan KASLR pada instans (khusus Ubuntu)

Untuk menjalankan hibernasi pada instans yang baru diluncurkan dengan Ubuntu 16.04 LTS (Xenial Xerus), Ubuntu 18.04 LTS (Bionic Beaver) yang dirilis dengan nomor seri 20190722.1 atau setelahnya, atau Ubuntu 20.04 LTS (Focal Fossa) yang dirilis dengan nomor seri 20210820 atau setelahnya, kami sarankan untuk menonaktifkan KASLR (Kernel Address Space Randomization). Di Ubuntu 16.04 LTS, Ubuntu 18.04 LTS, atau Ubuntu 20.04 LTS, KASLR diaktifkan secara default.

KASLR adalah fitur keamanan kernel Linux standar yang membantu mengurangi paparan dan konsekuensi dari kerentanan akses memori yang belum ditemukan dengan mengacak nilai alamat

dasar kernel. Dengan KASLR diaktifkan, ada kemungkinan bahwa instans tidak dapat dilanjutkan setelah hibernasi.

Untuk mempelajari selengkapnya tentang KASLR, lihat Fitur Ubuntu.

Untuk menonaktifkan KASLR pada instans yang diluncurkan dengan Ubuntu

1. Hubungkan ke instans Anda menggunakan SSH. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..
2. Buka file `/etc/default/grub.d/50-cloudimg-settings.cfg` dengan editor pilihan Anda. Edit baris `GRUB_CMDLINE_LINUX_DEFAULT` untuk menambahkan opsi `nokaslr` ke akhirnya, seperti yang ditunjukkan pada contoh berikut.

```
GRUB_CMDLINE_LINUX_DEFAULT="console=tty1 console=ttyS0
 nvme_core.io_timeout=4294967295 nokaslr"
```

3. Simpan file dan keluar dari editor Anda.
4. Jalankan perintah berikut untuk membangun ulang konfigurasi grub.

```
sudo update-grub
```

5. Boot ulang instans.

```
sudo reboot
```

6. Jalankan perintah berikut untuk mengonfirmasi bahwa `nokaslr` telah ditambahkan.

```
cat /proc/cmdline
```

   Output dari perintah harus menyertakan opsi `nokaslr`.

## Hibernasi instans Amazon EC2

Anda dapat memulai hibernasi pada Inst On-Demand ans atau Instans Spot jika instans tersebut adalah EBS-backed instance, diaktifkan untuk hibernasi, dan memenuhi prasyarat hibernasi. Jika sebuah instans tidak berhasil melakukan hibernasi, pematian normal akan terjadi.

Console

Untuk menghibernasi instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih sebuah instans, dan pilih instans state, Hibernate instans. Jika instans Hibernasi dinonaktifkan, instans tersebut sudah hibernasi atau dihentikan, atau tidak dapat dihibernasi. Untuk informasi selengkapnya, lihat Prasyarat untuk hibernasi instans EC2.
4. Ketika diminta konfirmasi, pilih Hibernasi. Perlu waktu beberapa menit agar instans mengalami hibernasi. Status instans pertama berubah menjadi Berhenti, lalu berubah menjadi Berhenti saat instans telah hibernasi.

AWS CLI

Untuk menghibernasi instans

Gunakan perintah stop-instances dan tentukan parameter `--hibernate`.

```
aws ec2 stop-instances \
    --instance-ids i-1234567890abcdef0 \
    --hibernate
```

PowerShell

Untuk menghibernasi instans

Gunakan Stop-EC2Instance cmdlet.

```
Stop-EC2Instance `
    -InstanceId i-1234567890abcdef0 `
    -Hibernate $true
```

Anda dapat memeriksa apakah hibernasi dimulai pada instance.

Console

Untuk melihat apakah hibernasi dimulai pada sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih instans dan, pada tab Detail, di bagian Detail instans, periksa nilai untuk Pesan transisi status.

   Client.UserInitiatedHibernate: Hibernasi yang diprakarsai pengguna menunjukkan bahwa Anda memulai hibernasi pada Instance atau On-Demand Instance Spot.

AWS CLI

Untuk melihat apakah hibernasi dimulai pada sebuah instans

Gunakan perintah describe-instances dan tentukan filter `state-reason-code` untuk melihat instans tempat hibernasi diinisiasi.

```
aws ec2 describe-instances \
    --filters "Name=state-reason-code,Values=Client.UserInitiatedHibernate"
```

Bidang berikut dalam output menunjukkan bahwa hibernasi dimulai pada Instance atau On-Demand Instance Spot.

```
"StateReason": {
    "Code": "Client.UserInitiatedHibernate"
}
```

PowerShell

Untuk melihat apakah hibernasi dimulai pada sebuah instans

Gunakan Get-EC2Instance cmdlet dan tentukan `state-reason-code` filter untuk melihat instance di mana hibernasi dimulai.

```
Get-EC2Instance `
    -Filter @{Name="state-reason-code";Value="Client.UserInitiatedHibernate"}
```

## Memulai instans Amazon EC2 yang hibernasi

Mulai instans hibernasi dengan memulainya dengan cara yang sama seperti Anda memulai instans yang dihentikan.

Untuk Instans Spot, jika Amazon EC2 melakukan hibernasi instans, maka hanya Amazon EC2 yang dapat melanjutkannya. Anda hanya dapat melanjutkan Instans Spot yang hibernasi jika Anda menghibernasinya. Instans Spot hanya dapat dilanjutkan jika kapasitas tersedia dan harga Spot kurang dari atau sama dengan harga maksimum yang Anda tentukan.

Console

Untuk memulai instans yang dihibernasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans hibernasi, dan pilih Status instans, Mulai instans. Hal ini diperlukan waktu beberapa menit hingga instans memasuki status `running`. Selama waktu ini, pemeriksaan status instans menunjukkan instans dalam status gagal sampai instans dimulai.

AWS CLI

Untuk memulai instans yang dihibernasi

Gunakan perintah start-instances.

```
aws ec2 start-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk memulai instans yang dihibernasi

Gunakan Start-EC2Instance cmdlet.

```
Start-EC2Instance -InstanceId i-1234567890abcdef0
```

## Memecahkan masalah hibernasi instans Amazon EC2

Gunakan informasi ini untuk membantu mendiagnosis dan memperbaiki masalah yang mungkin Anda temui saat menghibernasi sebuah instans.

Masalah hibernasi

- Tidak dapat berhibernasi segera setelah peluncuran

- Butuh waktu terlalu lama untuk beralih dari berhenti kepada dihentikan, dan status memori tidak dipulihkan setelah mulai
- Contoh “macet” di berhenti status
- Tidak dapat memulai Instans Spot segera setelah hibernasi
- Gagal melanjutkan Instans Spot

Tidak dapat berhibernasi segera setelah peluncuran

Jika Anda mencoba untuk menghibernasi sebuah instans terlalu cepat setelah Anda meluncurkannya, Anda mendapatkan pesan kesalahan.

Anda harus menunggu sekitar dua menit untuk instance Linux dan sekitar lima menit untuk instance Windows setelah peluncuran sebelum hibernasi.

Butuh waktu terlalu lama untuk beralih dari berhenti kepada dihentikan, dan status memori tidak dipulihkan setelah mulai

Jika instans hibernasi Anda memerlukan waktu lama untuk bertransisi dari status `stopping` ke `stopped`, dan jika status memori tidak dipulihkan setelah Anda memulainya, ini mungkin menunjukkan bahwa hibernasi tidak dikonfigurasi dengan benar.

Instans Linux

Periksa log sistem instans dan cari pesan yang terkait dengan hibernasi. Untuk mengakses log sistem, hubungkan ke instans atau gunakan perintah get-console-output. Menemukan baris log dari `hibinit-agent`. Jika garis log menunjukkan kegagalan atau garis log hilang, kemungkinan besar ada kegagalan dalam mengonfigurasi hibernasi saat peluncuran.

Misalnya, pesan berikut menunjukkan bahwa volume root instans tidak cukup besar: `hibinit-agent: Insufficient disk space. Cannot create setup for hibernation. Please allocate a larger root device.`

Jika baris log terakhir dari `hibinit-agent` adalah `hibinit-agent: Running: swapoff / swap`, hibernasi berhasil dikonfigurasi.

Jika Anda tidak melihat log apa pun dari proses ini, AMI Anda mungkin tidak mendukung hibernasi. Untuk informasi tentang AMI yang didukung, lihat Prasyarat untuk hibernasi instans EC2. Jika Anda menggunakan AMI Linux Anda sendiri, pastikan Anda mengikuti instruksi untukKonfigurasikan AMI Linux untuk mendukung hibernasi.

Windows Server 2016 dan setelahnya

Periksa log peluncuran EC2 dan cari pesan yang terkait dengan hibernasi. Untuk mengakses log peluncuran EC2, hubungkan ke instans dan buka file `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\Ec2Launch.log` di editor teks. Jika Anda menggunakan EC2Launch v2, buka `C:\ProgramData\Amazon\EC2Launch\log\agent.log`.

> **Note**
>
> Secara default, Windows menyembunyikan file dan folder di bawah `C:\ProgramData`. Untuk melihat direktori dan file EC2Launch, masukkan jalur di Windows Explorer atau ubah properti folder untuk menampilkan file dan folder tersembunyi.

Temukan garis log untuk hibernasi. Jika garis log menunjukkan kegagalan atau garis log hilang, kemungkinan besar ada kegagalan dalam mengonfigurasi hibernasi saat peluncuran.

Misalnya, pesan berikut menunjukkan bahwa hibernasi gagal dikonfigurasi: `Message: Failed to enable hibernation.` Jika pesan kesalahan tersebut menyertakan nilai ASCII desimal, Anda dapat mengonversi nilai ASCII menjadi teks biasa untuk membaca pesan kesalahan lengkap.

Jika baris log berisi `HibernationEnabled: true`, hibernasi berhasil dikonfigurasi.

Windows Server 2012 R2 dan sebelumnya

Periksa log konfigurasi EC2 dan cari pesan yang terkait dengan hibernasi. Untuk mengakses log konfigurasi EC2, hubungkan ke instans dan buka file `C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2ConfigLog.txt` di editor teks. Temukan baris log untuk `SetHibernateOnSleep`. Jika baris log menunjukkan kegagalan atau baris log hilang, kemungkinan besar ada kegagalan dalam mengonfigurasi hibernasi saat peluncuran.

Misalnya, pesan berikut menunjukkan bahwa volume root instans tidak cukup besar: `SetHibernateOnSleep: Failed to enable hibernation: Hibernation failed with the following error: There is not enough space on the disk.`

Jika baris log adalah `SetHibernateOnSleep: HibernationEnabled: true`, hibernasi berhasil dikonfigurasi.

Ukuran instance Windows

Jika Anda menggunakan instance Windows T3 atau T3a dengan RAM kurang dari 1 GiB, coba tingkatkan ukuran instance menjadi instans yang memiliki setidaknya 1 GiB RAM.

Contoh “macet” di berhenti status

Jika Anda menghibernasi instans Anda dan instans tersebut tampak "macet" di status `stopping`, Anda dapat menghentikannya secara paksa. Untuk informasi selengkapnya, lihat Memecahkan masalah penghentian instans Amazon EC2.

Tidak dapat memulai Instans Spot segera setelah hibernasi

Jika Anda mencoba memulai instans Spot dalam waktu dua menit setelah hibernasi, Anda mungkin mendapatkan kesalahan berikut:

```
You failed to start the Spot Instance because the associated Spot Instance request is not in an appropriate state to support start.
```

Tunggu sekitar dua menit untuk instance Linux dan sekitar lima menit untuk instans Windows dan kemudian coba lagi memulai instance.

Gagal melanjutkan Instans Spot

Jika Instans Spot berhasil dihibernasi tetapi gagal dilanjutkan, dan sebagai gantinya di-boot ulang (restart baru di mana status hibernasi tidak dipertahankan), itu mungkin karena data pengguna berisi skrip berikut:

```
/usr/bin/enable-ec2-spot-hibernation
```

Hapus skrip ini dari bidang Data pengguna di templat peluncuran, lalu minta instans Spot baru.

Perhatikan bahwa meskipun instans gagal dilanjutkan, tanpa status hibernasi yang dipertahankan, instans masih dapat dimulai dengan cara yang sama seperti memulai dari status. `stopped`

## Mulai ulang instans Amazon EC2 Anda

Sebuah instans yang melakukan boot ulang setara dengan penyalaan ulang sistem operasi. Dalam kebanyakan kasus, hanya diperlukan beberapa menit untuk melakukan boot ulang instans Anda.

Saat Anda melakukan boot ulang sebuah instans, hal-hal berikut akan tetap:

- Nama DNS publik (IPv4)
- Alamat IPv4 privat
- Alamat IPv4 publik
- Alamat IPv6 (jika ada)
- Setiap data pada volume penyimpanan instansnya

Mem-boot ulang instans tidak memulai periode penagihan instans baru, tidak seperti menghentikan dan memulai instance (yang memulai periode penagihan baru dengan biaya minimum satu menit).

Reboot instans dapat dimulai oleh pengguna (di mana Anda me-reboot instance secara manual) atau diprakarsai oleh AWS (untuk pemulihan instans otomatis, atau sebagai tanggapan terhadap peristiwa reboot terjadwal untuk pemeliharaan yang diperlukan, seperti untuk menerapkan pembaruan yang memerlukan reboot).

Untuk reboot yang diprakarsai pengguna, sebaiknya gunakan konsol Amazon EC2, CLI, atau API alih-alih menjalankan perintah reboot sistem operasi dari instans Anda. Saat menggunakan Amazon EC2, jika instans tidak mati secara bersih dalam beberapa menit, Amazon EC2 melakukan hard reboot. Selanjutnya, AWS CloudTrail membuat catatan API saat instance Anda di-boot ulang.

Topik ini menjelaskan cara melakukan reboot yang diprakarsai pengguna. Untuk informasi tentang reboot yang dilakukan oleh AWS, lihat Pemulihan instans otomatis danKelola instans Amazon EC2 yang dijadwalkan untuk reboot.

> **Important**
>
> Jika pembaruan diinstal pada instans Anda, sebaiknya Anda tidak me-reboot atau mematikan instans menggunakan konsol Amazon EC2 atau baris perintah sampai semua pembaruan diinstal. Saat Anda menggunakan konsol Amazon EC2 atau baris perintah untuk melakukan boot ulang atau mematikan instans Anda, ada risiko bahwa instans Anda akan sulit di-boot ulang. Boot ulang paksa saat pembaruan sedang diinstal dapat membuat instans Anda menjadi tidak stabil.

Console

Untuk melakukan boot ulang instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih instans dan pilih instans state, Reboot instans.
4. Pilih Boot ulang ketika diminta untuk konfirmasi.

   Instans tetap dalam status `running`.

AWS CLI

Untuk melakukan boot ulang instans

Gunakan perintah reboot-instance.

```
aws ec2 reboot-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk melakukan boot ulang instans

Gunakan Restart-EC2Instance cmdlet.

```
Restart-EC2Instance -InstanceId i-1234567890abcdef0
```

Untuk menjalankan eksperimen injeksi kesalahan terkontrol

Anda dapat menggunakannya AWS Fault Injection Service untuk menguji bagaimana aplikasi Anda merespons saat instance Anda di-boot ulang. Untuk informasi selengkapnya, lihat Panduan Pengguna AWS Fault Injection Service.

## Menghentikan instans Amazon EC2

**Warning**

Mengakhiri instance bersifat permanen dan tidak dapat diubah.
Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.

Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Anda dapat menghapus instans Anda saat tidak lagi membutuhkannya. Hal ini disebut sebagai mengakhiri instans Anda. Segera setelah status instans berubah menjadi `shutting-down` atau `terminated`, Anda tidak lagi dikenai biaya untuk instans itu.

Anda tidak dapat terhubung ke atau memulai sebuah instans setelah mengakhirinya. Namun, Anda dapat meluncurkan instans baru menggunakan AMI yang sama.

Jika Anda lebih suka menghentikan atau melakukan hibernasi instans, lihat Menghentikan dan memulai instans Amazon EC2 atauHibernasi instans Amazon EC2 Anda. Untuk informasi selengkapnya, lihat Perbedaan antara status instance.

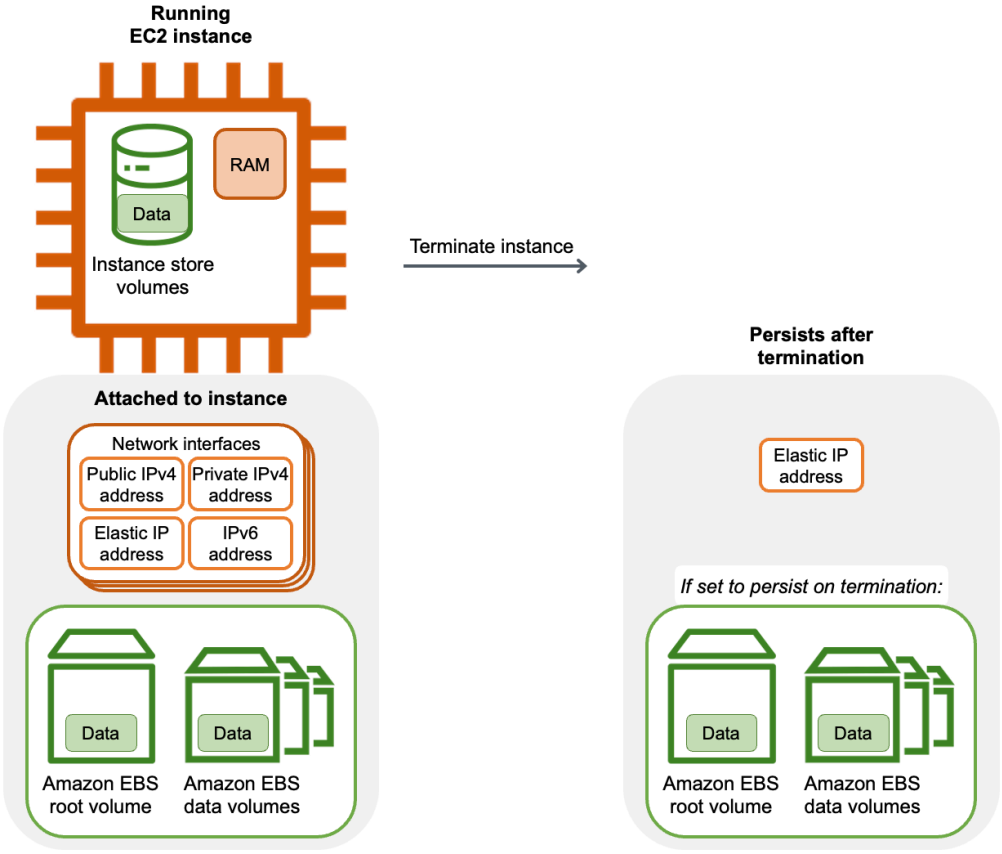
Daftar Isi


- Cara kerja penghentian instans
- Metode untuk mengakhiri instance
- Hentikan instance dengan shutdown OS yang anggun
- Hentikan instance dan lewati shutdown OS yang anggun
- Memecahkan masalah pengakhiran instans
- Ubah perlindungan penghentian instans
- Ubah perilaku shutdown yang diprakarsai instans
- Pertahankan data saat instans diakhiri


## Cara kerja penghentian instans

Saat Anda menghentikan instance, perubahan didaftarkan pada tingkat sistem operasi (OS) instans, beberapa sumber daya hilang, dan beberapa sumber daya tetap ada.

Diagram berikut menunjukkan apa yang hilang dan apa yang tetap ada saat instans Amazon EC2 dihentikan. Ketika instance dihentikan, data pada volume penyimpanan instans apa pun dan data yang disimpan dalam RAM instans dihapus. Setiap alamat IP Elastic yang terkait dengan instance dipisahkan. Untuk volume root dan volume data Amazon EBS, hasilnya tergantung pada  pengaturan H  apus pada penghentian setiap volume.




Pertimbangan-pertimbangan

- Ketekunan data
  - Volume penyimpanan instans: Semua data dihapus secara permanen saat instance dihentikan.
  - Volume akar EBS:
    - Saat dilampirkan saat peluncuran, dihapus secara default saat instance berakhir.
    - Saat dilampirkan setelah peluncuran, tetap ada secara default saat instance berakhir.
  - Volume data EBS:
    - Saat dilampirkan saat peluncuran menggunakan konsol: Bertahan secara default saat instance berakhir.
    - Saat dilampirkan saat peluncuran menggunakan CLI: Dihapus secara default saat instance berakhir.
    - Saat dilampirkan setelah peluncuran menggunakan konsol atau CLI: Bertahan secara default saat instance berakhir.

> **Note**
>
> Volume apa pun yang tidak dihapus pada penghentian instans terus dikenakan biaya. Anda dapat mengubah pengaturan sehingga volume dihapus atau tetap ada pada penghentian instans. Untuk informasi selengkapnya, lihat Pertahankan data saat instans diakhiri.

- Perlindungan terhadap penghentian yang tidak disengaja
  - Untuk mencegah instans dihentikan secara tidak sengaja oleh seseorang, aktifkan perlindungan penghentian.
  - Untuk mengontrol apakah instance berhenti atau dihentikan saat shutdown dimulai dari instans, ubah perilaku shutdown yang dimulai instans.
- Skrip shutdown — Jika Anda menjalankan skrip pada penghentian instans, instans Anda mungkin mengalami penghentian abnormal karena kami tidak memiliki cara untuk memastikan bahwa skrip shutdown berjalan. Amazon EC2 mencoba mematikan instans dengan bersih dan menjalankan skrip penutupan sistem apa pun; namun, peristiwa tertentu (seperti kegagalan perangkat keras) dapat mencegah skrip shutdown sistem ini berjalan.
- Instans bare metal — Instans bare metal x86 tidak mendukung shutdown kooperatif.

Hal yang terjadi ketika Anda mengakhiri sebuah instans

Perubahan terdaftar di tingkat OS

- Permintaan API akan mengirimkan peristiwa penekanan tombol kepada tamu.
- Berbagai layanan sistem dihentikan sebagai akibat dari peristiwa penekanan tombol. Shutdown sistem yang anggun disediakan oleh systemd (Linux) atau proses Sistem (Windows). Pematian yang tertib dipicu oleh peristiwa penekanan tombol pematian ACPI dari hypervisor.
- Pematian ACPI dimulai.
- Instans dimatikan setelah proses shutdown yang anggun keluar. Tidak ada waktu pematian OS yang dapat dikonfigurasi. Instans akan tetap terlihat di konsol untuk beberapa saat, kemudian entri tersebut akan dihapus secara otomatis.

Sumber daya hilang

- Data disimpan di volume penyimpanan instans.

- Volume root EBS jika `DeleteOnTermination` atribut disetel `ketrue`.
- Volume data EBS (dilampirkan saat peluncuran atau setelahnya) jika `DeleteOnTermination` atribut disetel `ketrue`.

Sumber daya yang bertahan

- Volume root EBS jika `DeleteOnTermination` atribut disetel `kefalse`.
- Volume data EBS (dilampirkan saat peluncuran atau setelahnya) jika `DeleteOnTermination` atribut disetel `kefalse`.

Uji respons aplikasi terhadap pengakhiran instans

Anda dapat menggunakannya AWS Fault Injection Service untuk menguji bagaimana aplikasi Anda merespons saat instance Anda dihentikan. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS Fault Injection Service.

## Metode untuk mengakhiri instance

> **Warning**
>
> Mengakhiri instance bersifat permanen dan tidak dapat diubah.
> Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
> Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Ada empat cara untuk melakukan penghentian instans yang diprakarsai pengguna: penghentian default, akhiri dengan skip OS shutdown, paksa terminasi, dan paksa akhiri dengan skip OS shutdown. Tabel berikut membandingkan perbedaan utama antara metode terminasi:

> **Note**
>
> Anda tidak dapat menghentikan instans jika perlindungan penghentian diaktifkan. Untuk informasi selengkapnya, lihat Meng ubah perlindungan penghentian instans.

| Metode penghentian | Tujuan utama | Kasus penggunaan | Perintah CLI |
|---|---|---|---|
| Default berhenti | Penghentian instans normal dengan upaya shutdown OS yang anggun. | Penghentian instance yang khas. | `aws ec2 terminate-instances \ --instance-id i-1234567890abcdef0` |
| Hentikan dengan skip OS shutdown | Melewati shutdown OS yang anggun saat mengakhiri instance. | Saat melewati penghentian OS yang anggun diperlukan. | `aws ec2 terminate-instances \ --instance-id i-1234567890abcdef0 \ --skip-os-shutdown` |
| Berhenti paksa | Menangani contoh macet. Mencoba penghentian default terlebih dahulu; jika instance gagal diakhiri, maka instans akan diakhiri secara paksa. | Ketika instance macet dalam `shutting-down` keadaan. | `aws ec2 terminate-instances \ --instance-id i-1234567890abcdef0 \ --force` |

| Metode penghentian | Tujuan utama | Kasus penggunaan | Perintah CLI |
| --- | --- | --- | --- |
| Paksa diakhiri dengan lewati penutupan OS | Force mengakhiri dan melewati shutdown OS yang anggun saat mengakhiri instance. | Ketika penghentian paksa dan melewati shutdown OS yang anggun diperlukan. | `aws ec2 terminate-instances \ --instance-id i-1234567890abcdef0 \ --force \ --skip-os-shutdown` |

Untuk petunjuk tentang cara menggunakan setiap metode, lihat berikut ini:

- Hentikan instance dengan shutdown OS yang anggun
- Hentikan instance dan lewati shutdown OS yang anggun
- Memaksa mengakhiri instance

## Hentikan instance dengan shutdown OS yang anggun

Anda dapat menghentikan instance menggunakan metode penghentian default, yang mencakup upaya shutdown OS yang anggun. Untuk informasi selengkapnya, lihat Metode untuk mengakhiri instance.

Console

Untuk mengakhiri instance menggunakan metode penghentian default

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance, dan pilih Status instans, H entikan (hapus) instance.
4. Pilih Hentikan (hapus) saat diminta konfirmasi.
5. Setelah Anda menghentikan instance, instans tetap terlihat untuk sementara waktu, dengan status `terminated`.

Jika penghentian gagal atau instans yang dihentikan terlihat selama lebih dari beberapa jam, lihatInstans yang dihentikan masih ditampilkan.

AWS CLI

Untuk mengakhiri instance menggunakan metode penghentian default

Gunakan perintah terminate-instance.

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk mengakhiri instance menggunakan metode penghentian default

Gunakan Remove-EC2Instance cmdlet.

```
Remove-EC2Instance -InstanceId i-1234567890abcdef0
```

## Hentikan instance dan lewati shutdown OS yang anggun

Anda dapat melewati shutdown OS yang anggun saat mengakhiri instance. Untuk informasi selengkapnya, lihat Metode untuk mengakhiri instance.

Console

Untuk menghentikan instance dan melewati shutdown OS yang anggun

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance, dan pilih Status instans, H entikan (hapus) instance.
4. Di bawah Lewati penutupan OS, pilih kotak centang Lewati penutupan OS. Jika Anda tidak melihat opsi ini di konsol, opsi ini belum tersedia di konsol di Wilayah saat ini. Namun, Anda dapat mengakses fitur ini menggunakan SDK AWS CLI atau, atau mencoba Wilayah lain di konsol.
5. Pilih Hentikan (hapus).

6. Setelah Anda menghentikan instance, instans tetap terlihat untuk sementara waktu, dengan status`terminated`.

   Jika penghentian gagal atau instans yang dihentikan terlihat selama lebih dari beberapa jam, lihatInstans yang dihentikan masih ditampilkan.

AWS CLI

Untuk mengakhiri instance dan melewati shutdown OS yang anggun

Gunakan perintah terminate-instance dengan.. `--skip-os-shutdown`

```
aws ec2 terminate-instances \
    --instance-ids i-1234567890abcdef0 \
    --skip-os-shutdown
```

PowerShell

Untuk mengakhiri instance dan melewati shutdown OS yang anggun

Gunakan Remove-EC2Instance cmdlet dengan.. `-SkipOsShutdown $true`

```
Remove-EC2Instance `
    -InstanceId i-1234567890abcdef0 `
    -SkipOsShutdown $true
```

## Memecahkan masalah pengakhiran instans

Pemohon harus memiliki izin untuk menelep `ec2:TerminateInstances` on. Untuk informasi selengkapnya, lihat Contoh kebijakan untuk bekerja dengan instance.

Jika Anda mengakhiri instans Anda dan instans lain dimulai, kemungkinan besar Anda telah mengonfigurasi penskalaan otomatis melalui fitur seperti Armada EC2 atau Amazon EC2 Auto Scaling. Untuk informasi selengkapnya, lihat Instans diluncurkan atau dihentikan secara otomatis.

> **Note**
>
> Anda tidak dapat menghentikan instans jika perlindungan penghentian diaktifkan. Untuk informasi selengkapnya, lihat Meng ubah perlindungan penghentian instans.

Jika instance Anda berada dalam `shutting-down` status lebih lama dari biasanya, Anda dapat mencoba menghentikannya secara paksa. Jika tetap di `shutting-down` negara bagian, itu harus dibersihkan (dihentikan) oleh proses otomatis dalam layanan Amazon EC2. Untuk informasi selengkapnya, lihat Penghentian instans yang tertunda.

## Ubah perlindungan penghentian instans

Untuk mencegah instans Anda dihentikan secara tidak sengaja menggunakan Amazon EC2 API, baik Anda menelepon `TerminateInstances` langsung atau menggunakan antarmuka lain seperti konsol Amazon EC2, aktifkan perlindungan penghentian untuk instans tersebut. A `DisableApiTermination` tribut mengontrol apakah instance dapat dihentikan. Secara default, perlindungan pengakhiran dinonaktifkan untuk instans Anda. Anda dapat mengatur nilai atribut ini saat meluncurkan instance, atau saat instance sedang berjalan atau dihentikan.

A `DisableApiTermination` tribut tidak mencegah Anda menghentikan instance dengan memulai shutdown dari instance (misalnya, dengan menggunakan perintah sistem operasi untuk penutupan sistem) saat `InstanceInitiatedShutdownBehavior` atribut disetel `keterminate`. Untuk informasi selengkapnya, lihat Ubah perilaku shutdown yang diprakarsai instans.

Pertimbangan

- Mengaktifkan perlindungan penghentian tidak AWS mencegah penghentian instans ketika ada acara ter jadwal untuk mengakhiri instans.
- Mengaktifkan perlindungan pengakhiran tidak mencegah Amazon EC2 Auto Scaling untuk mengakhiri instans saat instans tidak dalam kondisi baik atau selama peristiwa penskalaan ke dalam. Anda dapat mengontrol apakah grup Auto Scaling dapat mengakhiri instans tertentu saat menskalakan menggunakan perlindungan penskalaan ke dalam instans. Anda dapat mengontrol apakah grup Auto Scaling dapat mengakhiri instans yang tidak sehat dengan menangguhkan proses penskalaan ReplaceUnhealthy.
- Anda tidak dapat mengaktifkan perlindungan pengakhiran untuk Instans Spot.

Console

Untuk mengaktifkan perlindungan penghentian untuk instance saat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada dasbor, pilih Luncurkan instans.
3. Perluas detail lanjutan. Untuk perlindungan penghentian, pilih Aktifkan.

4. Ketika Anda selesai menentukan detail untuk instance Anda, pilih Luncurkan instance.

Untuk mengubah perlindungan penghentian untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih T indakan, Pengaturan instans, U bah perlindungan penghentian.
5. Untuk perlindungan Pengakhiran pilih atau hapus Aktifkan.
6. Pilih Simpan.

AWS CLI

Untuk mengaktifkan perlindungan penghentian untuk sebuah instance

Gunakan perintah modify- instance-atribut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --disable-api-termination
```

Untuk menonaktifkan perlindungan penghentian untuk sebuah instance

Gunakan perintah modify- instance-atribut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --no-disable-api-termination
```

PowerShell

Untuk mengaktifkan perlindungan penghentian untuk sebuah instance

Gunakan Edit-EC2InstanceAttribute cmdlet.

```
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -DisableApiTermination $true
```

Untuk menonaktifkan perlindungan penghentian untuk sebuah instance

Gunakan Edit-EC2InstanceAttribute cmdlet.

```
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -DisableApiTermination $false
```

Hentikan beberapa instance dengan perlindungan penghentian

Jika Anda menghentikan beberapa instans di beberapa Zona Ketersediaan dalam permintaan yang sama, dan satu atau beberapa instans tertentu diaktifkan untuk perlindungan penghentian, permintaan gagal dengan hasil berikut:

- Instans yang ditentukan yang berada dalam Zona Ketersediaan yang sama dengan instans yang dilindungi tidak diakhiri.
- Instans yang ditentukan yang berada di Zona Ketersediaan yang berbeda, di mana tidak ada instans yang ditentukan lainnya yang dilindungi, berhasil diakhiri.

Contoh

Misalkan Anda memiliki empat instance berikut di dua Zona Ketersediaan.

| Instans | Zona Ketersediaan | Perlindungan pengakhiran |
|---|---|---|
| Contoh 1 | A | `Disabled` |
| Contoh 2 | | `Disabled` |
| Contoh 3 | AZB | `Enabled` |
| Contoh 4 | | `Disabled` |

Jika Anda mencoba untuk mengakhiri semua instans ini dalam permintaan yang sama, maka permintaan tersebut akan melaporkan kegagalan dengan hasil sebagai berikut:

- Instans 1 dan Instance 2 berhasil dihentikan karena tidak ada instans yang diaktifkan untuk perlindungan penghentian.

- Instans 3 dan Instance 4 gagal diakhiri karena Instans 3 diaktifkan untuk perlindungan penghentian.

## Ubah perilaku shutdown yang diprakarsai instans

> **Warning**
>
> Mengakhiri instance bersifat permanen dan tidak dapat diubah.
> Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
> Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Saat Anda memulai pematian dari instans yang didukung Amazon EBS (menggunakan perintah seperti shutdown atau poweroff), instans akan berhenti secara default. Anda dapat mengubah perilaku ini sehingga instans berakhir dengan mengubah atribut `InstanceInitiatedShutdownBehavior` untuk instans. Anda dapat mengubah atribut ini saat instans sedang berjalan atau berhenti.

Perintah halt tidak memulai pematian. Jika digunakan, instans tidak diakhiri. Sebaliknya, instans menempatkan CPU ke `HLT` dan instans tersebut tetap berjalan.

> **Note**
>
> Atribut `InstanceInitiatedShutdownBehavior` hanya berlaku ketika Anda melakukan pematian dari sistem operasi instans itu sendiri. Ini tidak berlaku saat Anda menghentikan instans menggunakan API `StopInstances` atau konsol Amazon EC2.

### Console

Untuk mengubah perilaku pematian yang dinisiasi instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih Tindakan, Pengaturan instans, Ubah perilaku pematian.

   Perilaku pematian menampilkan perilaku saat ini.
5. Untuk mengubah perilaku, pada Perilaku pematian, pilih Hentikan atau Akhiri.
6. Pilih Simpan.

AWS CLI

Untuk mengubah perilaku pematian yang dinisiasi instans

Gunakan perintah modify- instance-atribut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --instance-initiated-shutdown-behavior terminate
```

PowerShell

Untuk mengubah perilaku pematian yang dinisiasi instans

Gunakan Edit-EC2InstanceAttribute cmdlet.

```
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -InstanceInitiatedShutdownBehavior terminate
```

## Pertahankan data saat instans diakhiri

Saat instans Amazon EC2 dihentikan, Anda dapat menyimpan data pada volume penyimpanan instans atau volume Amazon EBS. Topik ini menjelaskan cara memastikan data Anda tetap ada setelah penghentian instans.

Bagaimana penghentian instans memengaruhi volume root dan data

Volume penyimpanan instans

Ketika sebuah instance dihentikan, volume penyimpanan instans secara otomatis dihapus dan data hilang. Untuk menyimpan data ini di luar masa pakai instance, sebelum mengakhiri instance, salin data secara manual ke penyimpanan persisten, seperti volume Amazon EBS, bucket Amazon S3, atau sistem file Amazon EFS. Untuk informasi selengkapnya, lihat Opsi penyimpanan untuk instans Amazon EC2 Anda.

Volume Amazon EBS

Ketika sebuah instance dihentikan, volume EBS dihapus atau dipertahankan, tergantung pada nilai `DeleteOnTermination` atribut untuk setiap volume:

- Ya (konsol)/`true`(CLI) - Volume dihapus saat instance dihentikan.
- No (console)/`false`(CLI) — Volume dipertahankan saat instance dihentikan. Volume yang diawetkan terus dikenakan biaya.

> **Note**
>
> Setelah instance berakhir, Anda dapat mengambil snapshot dari volume yang diawetkan atau melampirkannya ke instance lain. Untuk menghindari dikenakan biaya, Anda harus menghapus volume.

Perilaku penghapusan default untuk volume EBS

`DeleteOnTermination`Nilai default berbeda tergantung pada jenis volume, apakah volume dilampirkan saat peluncuran atau setelahnya, dan metode (konsol atau CLI) yang digunakan untuk melampirkan volume:

| Tipe volume | Terlampir saat | Metode untuk melampirkan | Perilaku default pada penghentian instance |
|---|---|---|---|
| Volume root | Saat peluncuran | Konsol atau CLI | Delete |
| Volume root | Setelah peluncuran | Konsol atau CLI | Melestarikan |
| Volume data | Saat peluncuran | Konsol | Melestarikan |
| Volume data | Saat peluncuran | CLI | Delete |

| Tipe volume | Terlampir saat | Metode untuk melampirkan | Perilaku default pada penghentian instance |
|---|---|---|---|
| Volume data | Setelah peluncuran | Konsol dan CLI | Melestarikan |

Periksa pengaturan persistensi volume

Nilai default saat peluncuran untuk volume EBS ditentukan oleh `DeleteOnTermination` atribut yang ditetapkan pada AMI. Anda dapat mengubah nilai saat peluncuran instance, mengesampingkan pengaturan AMI. Sebaiknya verifikasi setelan default untuk `DeleteOnTermination` atribut setelah Anda meluncurkan instance.

Untuk memeriksa apakah volume Amazon EBS akan dihapus pada penghentian instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih tab Penyimpanan.
5. Di bawah Blokir perangkat, gulir ke kanan untuk memeriksa kolom Hapus pada penghentian.
   - Jika Ya, volume dihapus saat instance dihentikan.
   - Jika Tidak, volume tidak akan dihapus saat instance dihentikan. Volume apa pun yang tidak dihapus terus dikenakan biaya.

Ubah volume root untuk bertahan saat peluncuran

Anda dapat mengubah `DeleteOnTermination` atribut volume root EBS saat meluncurkan instance. Anda juga dapat menggunakan prosedur berikut untuk volume data.

Console

Untuk mengubah volume root instance agar tetap ada saat peluncuran

1. Ikuti prosedur untuk meluncurkan instans, tetapi jangan meluncurkan instans sampai Anda menyelesaikan langkah-langkah berikut guna mengubah volume root agar tetap ada.
2. Pada panel Konfigurasi penyimpanan, pilih L anjutan.
3. Di bawah volume EBS, perluas informasi volume root.

4. Untuk Hapus saat pengakhiran, pilih Tidak.
5. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

AWS CLI

Untuk mengubah volume root instance agar tetap ada saat peluncuran

Gunakan perintah run-instance untuk mengubah nilai `DeleteOnTermination` dalam pemetaan perangkat blok.

Tambahkan `--block-device-mappings` opsi:

```
--block-device-mappings file://mapping.json
```

Dalam `mapping.json`, tentukan nama perangkat, misalnya `/dev/sda1` atau `/dev/xvda`, dan untuk `DeleteOnTermination`, tentukan `false`.

```
[
  {
    "DeviceName": "device_name",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

PowerShell

Untuk mengubah volume root instance agar tetap ada saat peluncuran

Gunakan New-EC2Instance cmdlet untuk mengubah nilai `DeleteOnTermination` dalam pemetaan perangkat blok.

Tambahkan `-BlockDeviceMapping` opsi:

```
-BlockDeviceMapping $bdm
```

Dalam `bdm`, tentukan nama perangkat, misalnya `/dev/sda1` atau `/dev/xvda`, dan untuk `DeleteOnTermination`, tentukan `false`.

```
$ebd = New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice
$ebd.DeleteOnTermination = false
$bdm = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$bdm.DeviceName = "/dev/sda1"
$bdm.Ebs = $ebd
```

Ubah volume root dari instance yang sedang berjalan untuk bertahan

Anda dapat mengubah volume root EBS dari instance yang sedang berjalan untuk bertahan. Anda juga dapat menggunakan prosedur berikut untuk volume data.

AWS CLI

Untuk mengubah volume root agar bertahan

Gunakan perintah modify- instance-atribut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0  \
    --block-device-mappings file://mapping.json
```

Dalam `mapping.json`, tentukan nama perangkat, misalnya `/dev/sda1` atau `/dev/xvda`, dan untuk `--DeleteOnTermination`, tentukan `false`.

```
[
  {
    "DeviceName": "device_name",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

PowerShell

Untuk mengubah volume root agar bertahan

Gunakan Edit-EC2InstanceAttribute cmdlet.

Tambahkan `-BlockDeviceMapping` opsi:

```
-BlockDeviceMapping $bdm
```

Dalam `bdm`, tentukan nama perangkat, misalnya `/dev/sda1` atau `/dev/xvda`, dan untuk `DeleteOnTermination`, tentukan `false`.

```
$ebd = New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice
$ebd.DeleteOnTermination = false
$bdm = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$bdm.DeviceName = "/dev/sda1"
$bdm.Ebs = $ebd
```

## Pensiun instans

Sebuah instance dijadwalkan untuk dihentikan ketika AWS mendeteksi kegagalan yang tidak dapat diperbaiki dari perangkat keras yang mendasari yang menghosting instans. Jenis volume root instans menentukan perilaku penghentian instans:

- Jika volume root instans Anda adalah volume Amazon EBS, instans dihentikan, dan Anda dapat memulainya lagi kapan saja. Memulai instans yang dihentikan, migrasikan ke perangkat keras baru.
- Jika volume root instans Anda adalah volume penyimpanan instans, instance dihentikan, dan tidak dapat digunakan lagi.

Untuk informasi selengkapnya tentang tipe peristiwa instans, lihat Acara terjadwal untuk instans Amazon EC2.

Daftar Isi


- Identifikasi instans yang dijadwalkan untuk pensiun
- Tindakan yang harus diambil untuk EBS-backed contoh yang dijadwalkan untuk pensiun
- Tindakan yang harus diambil untuk instans yang didukung penyimpanan instans yang dijadwalkan untuk pensiun

## Identifikasi instans yang dijadwalkan untuk pensiun

Jika instans Anda dijadwalkan untuk pensiun, Anda akan menerima email sebelum peristiwa itu disertai dengan ID instans dan tanggal pensiun. Anda juga dapat memeriksa contoh yang dijadwalkan untuk pensiun.

> **Important**
>
> Jika instance dijadwalkan untuk pensiun, sebaiknya Anda mengambil tindakan sesegera mungkin, karena instans mungkin sudah tidak dapat dijangkau. Untuk informasi selengkapnya, lihat Check if your instance is reachable.

Opsi untuk mengidentifikasi contoh yang dijadwalkan untuk pensiun

- Pantau email untuk kontak akun
- Periksa instans Anda

Pantau email untuk kontak akun

Jika instance dijadwalkan untuk pensiun, kontak utama untuk akun dan kontak operasi menerima email sebelum acara. Email ini menyertakan ID instans dan tanggal pensiun yang dijadwalkan. Untuk informasi selengkapnya, lihat Memper barui kontak utama untuk AWS akun Anda dan Memper barui kontak alternatif untuk AWS akun Anda di Panduan AWS Account Management Referensi.

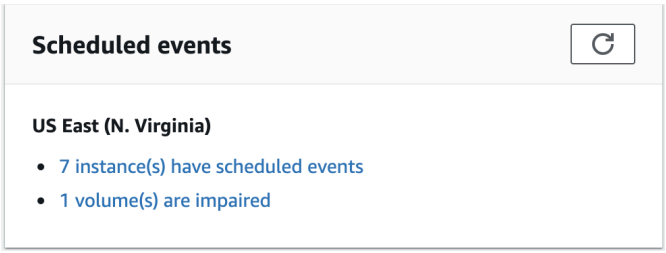
Periksa instans Anda

Jika Anda menggunakan akun email yang tidak Anda periksa secara teratur, Anda mungkin melewatkan pemberitahuan pensiun instans. Anda dapat memeriksa apakah ada instans Anda dijadwalkan untuk pensiun kapan saja.

Console

Untuk mengidentifikasi contoh yang dijadwalkan untuk pensiun

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor EC2. Di bawah Peristiwa terjadwal, Anda dapat melihat peristiwa yang terkait dengan instans dan volume Amazon EC2 Anda, yang diatur menurut Wilayah.




3. Jika Anda memiliki instans dengan peristiwa terjadwal yang terdaftar, pilih tautannya di bawah nama Wilayah untuk membuka halaman Peristiwa.
4. Halaman Peristiwa mencantumkan semua sumber daya yang memiliki peristiwa yang terkait dengannya. Untuk melihat instans yang dijadwalkan untuk pensiun, pilih sumber daya instans dari daftar filter pertama, kemudian instans atau pensiun dari daftar filter kedua.
5. Jika hasil filter menunjukkan bahwa sebuah instans dijadwalkan untuk pensiun, pilih instans itu, dan catat tanggal serta waktu di bidang Waktu mulai di panel detail. Ini adalah tanggal pensiun instans Anda.

AWS CLI

Untuk menemukan contoh yang dijadwalkan untuk pensiun

Gunakan perintah describe-inst ance-status berikut. Ulangi di setiap Wilayah tempat Anda menjalankan instance.

```
aws ec2 describe-instance-status --filters Name=event.code,Values=instance-
retirement
```

PowerShell

Untuk menemukan contoh yang dijadwalkan untuk pensiun

Gunakan Get-EC2InstanceStatus cmdlet. Ulangi di setiap Wilayah tempat Anda menjalankan instance.

```
Get-EC2InstanceStatus -Filter @{Name="event.code"; Values="instance-retirement"}
```

## Tindakan yang harus diambil untuk EBS-backed contoh yang dijadwalkan untuk pensiun

Untuk menyimpan data pada instans Anda yang pensiun, Anda dapat melakukan salah satu dari tindakan berikut. Anda harus mengambil tindakan ini sebelum tanggal pensiun instans untuk mencegah waktu henti dan kehilangan data yang tidak terduga.

Untuk instans Linux, jika Anda tidak yakin apakah instance Anda didukung oleh EBS atau penyimpanan instans, lihatVolume root untuk instans Amazon EC2 Anda.

### Periksa apakah instans Anda dapat dijangkau

Saat Anda mendapat notifikasi bahwa instans Anda dijadwalkan untuk pensiun, kami menyarankan agar Anda mengambil tindakan berikut secepat mungkin:

- Periksa apakah instans Anda dapat dijangkau dengan menghubungkan atau melakukan ping ke instans Anda.
- Jika instans Anda dapat dijangkau, stop/start Anda harus merencanakan instans pada waktu yang tepat sebelum tanggal pensiun yang dijadwalkan, ketika dampaknya minimal. Untuk informasi selengkapnya tentang menghentikan dan memulai instans Anda, dan apa yang akan terjadi saat instans Anda dihentikan, seperti efek pada alamat IP publik, privat, dan Elastis yang terkait dengan instans Anda, lihat Menghentikan dan memulai instans Amazon EC2. Perhatikan bahwa data pada volume penyimpanan instans hilang saat Anda menghentikan dan memulai instans Anda.
- Jika instans Anda tidak dapat dijangkau, Anda harus segera mengambil tindakan dan melakukan a stop/start untuk memulihkan instans Anda.
- Atau, jika Anda ingin mengakhiri instans Anda, rencanakan untuk melakukannya sesegera mungkin agar Anda tidak lagi dikenai biaya untuk instans tersebut.

### Buat cadangan instans Anda

Buat EBS-backed AMI dari instance Anda sehingga Anda memiliki cadangan. Untuk memastikan integritas data, hentikan instans sebelum Anda membuat AMI. Anda dapat menunggu tanggal pensiun yang dijadwalkan saat instans dihentikan, atau hentikan sendiri instans tersebut sebelum tanggal pensiun. Anda dapat memulai kembali instans kapan saja. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.

### Luncurkan instans pengganti

Setelah Anda membuat AMI dari instans, Anda dapat menggunakan AMI untuk meluncurkan instans pengganti. Dari konsol Amazon EC2, pilih AMI baru Anda, lalu pilih  Luncurkan instance dari AMI. Konfigurasikan parameter untuk instance Anda, lalu pilih  Luncurkan instance. Untuk informasi lebih lanjut tentang setiap bidang, lihatLuncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

## Tindakan yang harus diambil untuk instans yang didukung penyimpanan instans yang dijadwalkan untuk pensiun

Untuk menyimpan data pada instans Anda yang pensiun, Anda dapat melakukan salah satu dari tindakan berikut. Anda harus mengambil tindakan ini sebelum tanggal pensiun instans untuk mencegah waktu henti dan kehilangan data yang tidak terduga.

> **Warning**
>
> Jika instance Anda memiliki volume root penyimpanan instans dan melewati tanggal pensiunnya, instans akan dihentikan dan Anda tidak dapat memulihkan instans atau data apa pun yang disimpan di dalamnya. Terlepas dari jenis volume root instans Anda, data pada volume penyimpanan instans hilang saat instans dihentikan, bahkan jika volume dilampirkan ke instance dengan volume root EBS.

### Periksa apakah instans Anda dapat dijangkau

Saat Anda mendapat notifikasi bahwa instans Anda dijadwalkan untuk pensiun, kami menyarankan agar Anda mengambil tindakan berikut secepat mungkin:

- Periksa apakah instans Anda dapat dijangkau dengan menghubungkan atau melakukan ping ke instans Anda.
- Jika instans Anda tidak dapat dijangkau, kemungkinan sangat sedikit yang dapat dilakukan untuk memulihkan instans Anda. Untuk informasi lebih lanjut, lihatMemecahkan masalah instans Amazon EC2 yang tidak dapat dijangkau. AWS akan menghentikan instans Anda pada tanggal pensiun yang dijadwalkan, jadi, untuk instans yang tidak dapat dijangkau, Anda dapat segera menghentikan inst ans sendiri.

### Luncurkan instans pengganti

Buat Amazon S3-backed AMI dari instans Anda menggunakan alat AMI, seperti yang dijelaskan dalamBuat S3-backed AMI Amazon. Dari konsol Amazon EC2, pilih AMI baru Anda, lalu pilih Luncurkan instance dari AMI. Konfigurasikan parameter untuk instance Anda, lalu pilih Luncurkan instance. Untuk informasi lebih lanjut tentang setiap bidang, lihatLuncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

Mengonversi instance Anda menjadi EBS-backed instance

Transfer data Anda ke volume EBS, ambil snapshot dari volume tersebut, lalu buat AMI dari snapshot. Anda dapat meluncurkan instans pengganti dari AMI baru Anda. Lihat informasi yang lebih lengkap di Konversikan Amazon S3-backed AMI Anda menjadi EBS-backed AMI.

## Pemulihan instans otomatis

> **Important**
>
> Bagian ini menjelaskan cara mengkonfigurasi mekanisme pemulihan secara proaktif pada instance EC2. Mekanisme pemulihan ini dirancang untuk memulihkan ketersediaan instans saat AWS mendeteksi masalah perangkat keras atau perangkat lunak yang mendasari yang menyebabkan pemeriksaan status sistem gagal. Jika saat ini Anda mengalami masalah saat mengakses instans, lihat Memecahkan masalah inst ans EC2.

Jika AWS mendeteksi bahwa instance tidak tersedia karena masalah perangkat keras atau perangkat lunak yang mendasarinya, ada dua mekanisme yang dapat memulihkan ketersediaan instans secara otomatis — pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatch tindakan Amazon. Memulihkan ketersediaan instans juga dikenal sebagai pemulihan instans.

Selama proses pemulihan instans, AWS akan mencoba memindahkan instance Anda dari host dengan masalah perangkat keras atau perangkat lunak yang mendasarinya ke host lain. Jika berhasil, proses pemulihan instans akan muncul ke instance sebagai reboot yang tidak direncanakan. Anda dapat memverifikasi apakah pemulihan instans terjadi.

Jika proses pemulihan tidak berhasil, instance mungkin terus berjalan di host dengan masalah perangkat keras atau perangkat lunak yang mendasarinya. Dalam hal ini, diperlukan intervensi manual. Jika instans menjadi tidak dapat dijangkau atau pemeriksaan status sistem terus gagal, sebaiknya h entikan dan memulai instans secara manual. Ketika Anda memulai sebuah instance, biasanya dimigrasikan ke komputer host baru yang mendasarinya. Namun, tidak seperti pemulihan

instans otomatis, di mana instance mempertahankan alamat IPv4 publiknya, instance yang dimulai ulang menerima alamat IPv4 publik baru kecuali memiliki alamat IP Elastic.

Untuk mendapatkan manfaat dari mekanisme pemulihan otomatis, mekanisme tersebut harus dikonfigurasi terlebih dahulu pada instance sebelum pemeriksaan status sistem gagal. Secara default, pemulihan otomatis yang disederhanakan diaktifkan selama peluncuran instance. Anda dapat secara opsional mengonfigurasi pemulihan berbasis CloudWatch tindakan Amazon setelah peluncuran. Memiliki salah satu mekanisme ini dikonfigurasi membuat instance Anda lebih tangguh.

Pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatch tindakan Amazon hanya tersedia pada instans yang didukung. Untuk informasi selengkapnya, lihat Persyaratan untuk mengaktifkan pemulihan otomatis yang disederhanakan dan Persyaratan untuk mengaktifkan pemulihan berbasis CloudWatch tindakan.

> **Warning**
>
> Saat AWS memulihkan instance Anda karena masalah perangkat keras atau perangkat lunak yang mendasarinya, perhatikan konsekuensi berikut: data yang disimpan dalam memori volatile (RAM) akan hilang dan waktu aktif sistem operasi akan dimulai dari nol. Selain itu, dengan pemulihan berbasis CloudWatch tindakan, data pada volume penyimpanan instance juga akan hilang. Untuk membantu melindungi dari kehilangan data, kami menyarankan Anda membuat cadangan data berharga secara teratur. Untuk informasi selengkapnya tentang praktik terbaik pencadangan dan pemulihan untuk instans EC2, lihat Prakti k terbaik untuk Amazon EC 2.
>
> Mekanisme pemulihan instans otomatis dirancang untuk instance individual. Untuk panduan membangun sistem yang tangguh, lihatMembangun sistem yang tangguh.

Topik


- Konsep kunci pemulihan instans otomatis
- Perbedaan antara pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatch tindakan
- Membangun sistem yang tangguh
- Verifikasi apakah pemulihan instans otomatis terjadi
- Mengkonfigurasi pemulihan otomatis yang disederhanakan pada instans Amazon EC2
- Mengonfigurasi pemulihan berbasis CloudWatch tindakan pada instance EC2

# Konsep kunci pemulihan instans otomatis

Pemulihan instans otomatis adalah fitur Amazon EC2 yang secara otomatis mengembalikan ketersediaan instans saat terjadi kegagalan perangkat keras atau perangkat lunak yang mendasarinya, meningkatkan ketahanan dan keandalan instans EC2 Anda.

Berikut ini adalah konsep kunci pemulihan instans otomatis:

Opsi konfigurasi

Dua mekanisme dapat dikonfigurasi untuk mendukung pemulihan instans otomatis:

- Pemulihan otomatis yang disederhanakan: Diaktifkan secara default pada instans yang didukung.
- CloudWatch pemulihan berbasis tindakan: Memerlukan konfigurasi manual pada instans yang didukung.

Pemeriksaan status sistem

Pemeriksaan status sistem secara otomatis memantau AWS infrastruktur tempat instans EC2 Anda berjalan.

- Jika pemeriksaan status sistem gagal, AWS memulai pemulihan instans otomatis, yang mencoba memigrasikan instance yang terpengaruh ke perangkat keras yang berbeda.
- Pemeriksaan status sistem yang gagal menunjukkan masalah dengan perangkat keras atau perangkat lunak host, dan bukan masalah dengan instance itu sendiri. Pemulihan instans otomatis dapat memulihkan instance yang gagal dalam pemeriksaan status sistem. Namun, pemulihan instans otomatis tidak beroperasi jika hanya pemeriksaan status instance yang gagal.
- Untuk perbedaan antara pemeriksaan status instans dan sistem, lihat Jenis pemeriksaan status.

Contoh masalah perangkat keras atau perangkat lunak yang mendasarinya

Masalah perangkat keras atau perangkat lunak yang dapat menyebabkan pemeriksaan status sistem gagal termasuk hilangnya konektivitas jaringan, hilangnya daya sistem, masalah perangkat lunak pada host fisik, dan masalah perangkat keras pada host fisik yang memengaruhi jangkauan jaringan.

Karakteristik instance yang dipulihkan

Instance yang dipulihkan identik dengan instance asli, kecuali untuk elemen yang hilang.

Elemen yang diawetkan:

- ID Instans
- Alamat IP publik, pribadi, dan elastis
- Metadata instans
- Grup penempatan
- Volume EBS terlampir
- Zona Ketersediaan

Elemen yang hilang:

- Data disimpan dalam memori volatile (RAM)
- Data yang disimpan pada volume penyimpanan instans (hanya berlaku untuk pemulihan berbasis CloudWatch tindakan)
- Uptime sistem operasi diatur ulang ke nol

Pemeriksaan status sistem pemantauan dengan CloudWatch

Metrik StatusCheckFailed _System di CloudWatch menunjukkan apakah pemeriksaan status sistem lulus atau gagal.

Nilai metrik:

- 0 — Pemeriksaan status sistem lulus.
- 1 — Pemeriksaan status sistem gagal.

Acara di Dasbor Health

Selama upaya pemulihan instans otomatis, AWS mengirimkan peristiwa ke Anda Dasbor Health berdasarkan mekanisme pemulihan yang dikonfigurasi dan hasilnya:

- Pemulihan otomatis yang disederhanakan
  - Acara sukses: `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_SUCCESS`
  - Peristiwa kegagalan: `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_FAILURE`
- CloudWatch pemulihan berbasis tindakan
  - Acara sukses: `AWS_EC2_INSTANCE_AUTO_RECOVERY_SUCCESS`
  - Peristiwa kegagalan: `AWS_EC2_INSTANCE_AUTO_RECOVERY_FAILURE`

# Perbedaan antara pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatch tindakan

Tabel berikut membandingkan perbedaan utama antara pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatch tindakan.

| Titik perbandingan | Pemulihan otomatis yang disederhanakan | CloudWatch pemulihan berbasis tindakan |
|---|---|---|
| Konfigurasi | Diaktifkan secara default pada instans yang didukung | Membutuhkan konfigura si CloudWatch alarm dan tindakan manual |
| Fleksibilitas | Perilaku pemulihan tetap dikelola oleh AWS | Tindakan dan kondisi yang dapat disesuaikan |
| Notifikasi | Pemberitahuan dasar melalui Dasbor Health | Pemberitahuan yang dapat disesuaikan melalui SNS |
| Ukuran contoh logam | Dikecualikan | Termasuk |
| Volume penyimpanan instans terpasang saat peluncuran | Tidak didukung untuk instans yang melampirkan volume penyimpanan instans saat diluncurkan | Didukung pada jenis instance yang dipilih. Perhatikan bahwa data pada volume penyimpan an instans hilang selama pemulihan instans. |
| Waktu pemulihan | Upaya pemulihan standar | Upaya pemulihan lebih cepat daripada pemulihan otomatis yang disederhanakan |
| Masalah host diselesaikan selama migrasi | Migrasi mungkin dibatalkan dan instans tetap berada di host asli | Migrasi mungkin dibatalkan dan instans tetap berada di host asli |
| Biaya | Tidak ada biaya tambahan | Mungkin dikenakan biaya CloudWatch |

## Membangun sistem yang tangguh

Meskipun pemulihan otomatis yang disederhanakan dan pemulihan berbasis CloudWatch tindakan efektif untuk mempertahankan ketersediaan instans individual, AWS merekomendasikan penerapan arsitektur ketersediaan tinggi yang memungkinkan failover lalu lintas ke instans yang sehat.

Untuk mencapai hal ini, pertimbangkan untuk menggunakan AWS layanan seperti Elastic Load Balancing (yang mendistribusikan lalu lintas masuk ke beberapa instans EC2) dan Amazon EC2 Auto Scaling (yang secara otomatis menyesuaikan jumlah instans berdasarkan permintaan dan kesehatan).

Untuk informasi selengkapnya tentang membangun sistem yang tangguh dan toleran kesalahan dengan instans EC2, lihat sumber daya berikut:

- Kembali ke Dasar: Merancang untuk Kegagalan dengan EC2 di saluran AWS YouTube
- Arsitektur Pemulihan Bencana (DR) AWS, Bagian I: Strategi Pemulihan di Cloud di situs Blog AWS Arsitektur
- Panduan Pengguna Penyeimbang Beban Aplikasi
- Panduan Pengguna Penskalaan Otomatis Amazon EC2
- REL11-BP02 Gagal beralih ke sumber daya yang sehat dalam Ker AWS Well-Architected angka Pil ar Keandalan

## Verifikasi apakah pemulihan instans otomatis terjadi

Jika instans Anda tampaknya offline dan kemudian di-boot ulang secara tidak terduga, instans mungkin telah mengalami pemulihan instans otomatis sebagai respons terhadap masalah perangkat keras atau perangkat lunak yang mendasarinya. Anda dapat memverifikasi ini dengan memeriksa peristiwa pemulihan instans otomatis di Dasbor Health. Anda juga dapat memeriksa apakah masalah perangkat keras atau perangkat lunak yang mendasarinya terdeteksi untuk instance Anda dengan memeriksa CloudWatch metrik StatusCheckFailed _System Amazon.

### Periksa acara di Dasbor Health

Ketika upaya pemulihan instans otomatis terjadi, AWS mengirimkan peristiwa ke Anda Dasbor Health. Peristiwa spesifik tergantung pada mekanisme pemulihan yang dikonfigurasi dan apakah upaya berhasil atau gagal.

Untuk memeriksa peristiwa pemulihan instans otomatis di Dasbor Health

1. Buka Dasbor Health di https://phd.aws.amazon.com/phd/home #/.
2. Cari peristiwa yang terkait dengan pemulihan instans otomatis. Kehadiran peristiwa ini dapat mengkonfirmasi apakah upaya pemulihan instans otomatis terjadi dan hasilnya.
   - Pemulihan otomatis yang disederhanakan
     - Acara sukses: `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_SUCCESS`
     - Peristiwa kegagalan: `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_FAILURE`
   - CloudWatch pemulihan berbasis tindakan
     - Acara sukses: `AWS_EC2_INSTANCE_AUTO_RECOVERY_SUCCESS`
     - Peristiwa kegagalan: `AWS_EC2_INSTANCE_AUTO_RECOVERY_FAILURE`

## Memantau pemeriksaan status sistem dengan CloudWatch

Anda dapat memverifikasi apakah masalah perangkat keras atau perangkat lunak yang mendasari terdeteksi untuk instance Anda dengan memeriksa metrik StatusCheckFailed _System di CloudWatch. Nilai metrik menunjukkan apakah pemeriksaan status sistem lulus (tidak ada masalah perangkat keras atau perangkat lunak) atau gagal (masalah perangkat keras atau perangkat lunak).

Untuk memverifikasi apakah masalah perangkat keras atau perangkat lunak yang mendasarinya terdeteksi

1. Buka halaman Metrik CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/home? #metricsV2.
2. Verifikasi bahwa Anda berada di Wilayah yang sama dengan instans EC2 Anda.
3. Tempelkan metrik berikut di bidang pencarian Metrik, dan tekan Enter.

   ```
   StatusCheckFailed_System
   ```
4. Pilih EC2 > Per-Instance Metrik.
5. Dalam tabel, pilih kotak centang di sebelah instance yang ingin Anda centang.
6. Ubah periode kueri ke waktu di mana Anda mencurigai peristiwa pemulihan terjadi.
7. Pilih tab Graphed metrics, dan untuk StatusCheckFailed _System, lakukan hal berikut:

   a. Untuk Statistik, pilih Rat a-rata, Maksimum, atau Minimum.

b. Untuk Peri ode, pilih 1 menit.

8. Periksa nilai untuk StatusCheckFailed _System.

   - Nilai 0: Pemeriksaan status sistem lulus, menunjukkan tidak ada masalah perangkat keras atau perangkat lunak yang mendasarinya.
   - Nilai 1: Pemeriksaan status sistem gagal, menunjukkan masalah perangkat keras atau perangkat lunak yang mendasarinya.

Untuk informasi selengkapnya, lihat Pemulihan instans otomatis.

## Mengkonfigurasi pemulihan otomatis yang disederhanakan pada instans Amazon EC2

> **Important**
>
> Bagian ini menjelaskan cara mengkonfigurasi mekanisme pemulihan secara proaktif pada instance EC2. Mekanisme pemulihan ini dirancang untuk memulihkan ketersediaan instans saat AWS mendeteksi masalah perangkat keras atau perangkat lunak yang mendasari yang menyebabkan pemeriksaan status sistem gagal. Jika saat ini Anda mengalami masalah saat mengakses instans, lihat Memecahkan masalah inst ans EC2.

Jika AWS mendeteksi bahwa instance tidak tersedia karena masalah perangkat keras atau perangkat lunak yang mendasarinya, pemulihan otomatis yang disederhanakan dapat secara otomatis memulihkan ketersediaan instance dengan memindahkan instance dari host dengan masalah mendasar ke host lain.

Jika pemulihan otomatis yang disederhanakan AWS terjadi, kirim salah satu peristiwa berikut ke Anda Dasbor AWS Health, tergantung pada hasilnya:

- Acara sukses: `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_SUCCESS`
- Peristiwa kegagalan: `AWS_EC2_SIMPLIFIED_AUTO_RECOVERY_FAILURE`

Untuk diberitahu tentang peristiwa ini, Anda dapat mengonfigurasi notifikasi. Untuk informasi selengkapnya, lihat Membuat konfigurasi notifikasi pertama Anda Notifikasi Pengguna AWS di Panduan Notifikasi Pengguna AWS Pengguna. Anda juga dapat menggunakan EventBridge aturan Amazon untuk memantau peristiwa pemulihan otomatis yang disederhanakan.

Pemulihan otomatis yang disederhanakan diaktifkan secara default pada semua instans yang didukung selama peluncuran instance. Namun, itu hanya dapat beroperasi jika instance berada dalam `running` status, tidak ada peristiwa layanan yang tercantum di Dasbor AWS Health, dan ada kapasitas yang tersedia untuk jenis instans. Dalam beberapa situasi, seperti pemadaman yang signifikan, kendala kapasitas dapat menyebabkan upaya pemulihan gagal. Untuk informasi selengkapnya, lihat the section called “Memecahkan masalah kegagalan pemulihan otomatis yang disederhanakan”.

Anda dapat menonaktifkan pemulihan otomatis yang disederhanakan selama atau setelah peluncuran, dan mengaktifkannya kembali nanti jika diperlukan.

> **Warning**
>
> Saat AWS memulihkan instance Anda karena masalah perangkat keras atau perangkat lunak yang mendasarinya, perhatikan konsekuensi berikut: data yang disimpan dalam memori volatile (RAM) akan hilang dan waktu aktif sistem operasi akan dimulai dari nol. Untuk membantu melindungi dari kehilangan data, kami menyarankan Anda membuat cadangan data berharga secara teratur. Untuk informasi selengkapnya tentang praktik terbaik pencadangan dan pemulihan untuk instans EC2, lihat Prakti k terbaik untuk Amazon EC 2. Mekanisme pemulihan instans otomatis dirancang untuk instance individual. Untuk panduan membangun sistem yang tangguh, lihatMembangun sistem yang tangguh.

Daftar Isi


- Persyaratan untuk mengaktifkan pemulihan otomatis yang disederhanakan
- Konfigurasikan pemulihan otomatis yang disederhanakan
- Memecahkan masalah kegagalan pemulihan otomatis yang disederhanakan


## Persyaratan untuk mengaktifkan pemulihan otomatis yang disederhanakan

Pemulihan otomatis yang disederhanakan dapat diaktifkan pada instans yang memenuhi kriteria berikut:

Tipe instans

- Tujuan umum: A1, M3, M4, M5, M5a, M5n, M5zn, M6a, M6g, M6i, M6in, M7a, M7g, M7i,, M8a, M8azN, M8g, M8g, M8gn, M8i,, M8in, M8iB M7i-flex, M9g, T1, T2, T3, T3a, T4g M8i-flex

- Komputasi dioptimalkan: C  3, C4, C5, C5a, C5n, C6a, C6g, C6gn, C6i, C6in, C7a, C7g, C7gn, C7i,, C8a, C8g, C8Gn, C8i,, C8in, C8iN, C8iB, C9g C7i-flex C8i-flex
- Memori dioptimalkan:  R3, R4, R5, R5a, R5b, R5n, R6a, R6g, R6i, R6in, R7a, R7g, R7i, R7iz, R8a, R8g, R8g, R8g, R8i,, R8in, R8ib, R9g,,,,,,,,,,, X1, X1e, X2IEZn, X8g, X8i R8i-flex U-3tb1 U-6tb1 U-9tb1 U-12tb1 U-18tb1 U-24tb1 U7i-6tb U7i-8tb U7i-12tb U7in-16tb U7in-24tb U7in-32tb U7inh-32tb
- Komputasi yang dipercepat:  G3, G5g, Inf1, P3, VT1
- High-performance komputasi:  hpc6a, hpc7a, hpc7g, hpc8a

Penghunian

- Bersama
- Instans Khusus

  Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2.

Batasan

Pemulihan otomatis yang disederhanakan tidak didukung untuk contoh dengan karakteristik berikut:

- Ukuran instans: `metal` instance
- Penyewaan: Tuan Rumah Khusus. Untuk Host Khusus, gunakan Pemulihan Otom atis Host Khusus sebagai gantinya.
- Penyimpanan: Instans dengan volume penyimpanan instans
- Jaringan: Instans menggunakan Adaptor Kain Elastis
- Penskalaan Otomatis: Instans yang merupakan bagian dari grup Penskalaan Otomatis
- Pemeliharaan: Instans saat ini sedang menjalani acara pemeliharaan terjadwal

## Konfigurasikan pemulihan otomatis yang disederhanakan

Pemulihan otomatis yang disederhanakan diaktifkan secara default saat Anda meluncurkan instance yang didukung. Anda dapat mengatur perilaku pemulihan otomatis `disabled` selama atau setelah meluncurkan instance.

`default`Konfigurasi tidak mengaktifkan pemulihan otomatis yang disederhanakan untuk instance yang tidak didukung.

Console

Untuk menonaktifkan pemulihan otomatis simpel saat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans, kemudian pilih Luncurkan instans.
3. Di bagian Detail lanjutan, untuk pemulihan otomatis instans, pilih Din onaktifkan.
4. Konfigurasikan pengaturan peluncuran instans yang tersisa sesuai kebutuhan kemudian luncurkan instans.

Untuk menonaktifkan pemulihan otomatis yang disederhanakan setelah peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Tindakan, Pengaturan instans, Ubah perilaku pemulihan otomatis.
4. Pilih Nonaktif, lalu pilih Simpan.

Untuk mengaktifkan pemulihan otomatis yang disederhanakan setelah peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Tindakan, Pengaturan instans, Ubah perilaku pemulihan otomatis.
4. Pilih Default (Aktif), lalu pilih Simpan.

AWS CLI

Untuk menonaktifkan pemulihan otomatis simpel saat peluncuran

Gunakan perintah run-instance dengan opsi. `--maintenance-options`

```
--maintenance-options AutoRecovery=Disabled
```

Untuk menonaktifkan pemulihan otomatis yang disederhanakan setelah peluncuran

Gunakan perintah modify-instance-maintenance-options.

```
aws ec2 modify-instance-maintenance-options \
```

```
    --instance-id i-1234567890abcdef0 \
    --auto-recovery disabled
```

Untuk mengaktifkan pemulihan otomatis yang disederhanakan setelah peluncuran

Gunakan perintah modify-instance-maintenance-options.

```
aws ec2 modify-instance-maintenance-options \
    --instance-id i-1234567890abcdef0 \
    --auto-recovery default
```

PowerShell

Untuk menonaktifkan pemulihan otomatis simpel saat peluncuran

Gunakan New-EC2Instance cmdlet.

```
-MaintenanceOptions_AutoRecovery Disabled
```

Untuk menonaktifkan pemulihan otomatis yang disederhanakan setelah peluncuran

Gunakan Edit-EC2InstanceMaintenanceOption cmdlet.

```
Edit-EC2InstanceMaintenanceOption `
    -InstanceId i-1234567890abcdef0 `
    -AutoRecovery Disabled
```

Untuk mengaktifkan pemulihan otomatis yang disederhanakan setelah peluncuran

Gunakan Edit-EC2InstanceMaintenanceOption cmdlet.

```
Edit-EC2InstanceMaintenanceOption `
    -InstanceId i-1234567890abcdef0 `
    -AutoRecovery Enabled
```

## Memecahkan masalah kegagalan pemulihan otomatis yang disederhanakan

Jika pemulihan otomatis yang disederhanakan gagal memulihkan instance Anda, pertimbangkan masalah berikut:

- AWS acara layanan sedang berjalan

  Pemulihan otomatis yang disederhanakan tidak beroperasi selama acara layanan di Dasbor AWS Health. Anda mungkin tidak menerima notifikasi kegagalan pemulihan untuk peristiwa semacam itu. Untuk informasi ketersediaan layanan terbaru, lihat halaman Status kesehatan layanan.

- Kapasitas tidak mencukupi

  Perangkat keras pengganti sementara tidak mencukupi untuk memigrasikan instance.

- Upaya pemulihan harian maksimum tercapai

  Instance telah mencapai tunjangan harian maksimum untuk upaya pemulihan. Instans Anda mungkin akan dihentikan jika pemulihan otomatis gagal dan degradasi perangkat keras ditentukan sebagai akar penyebab pemeriksaan status sistem yang gagal.

Jika kegagalan pemeriksaan status sistem instans tetap ada meskipun ada beberapa upaya pemulihan, lihat Memec ahkan masalah instans dengan pemeriksaan status gagal untuk panduan tambahan.

## Mengonfigurasi pemulihan berbasis CloudWatch tindakan pada instance EC2

> **Important**
>
> Bagian ini menjelaskan cara mengkonfigurasi mekanisme pemulihan secara proaktif pada instance EC2. Mekanisme pemulihan ini dirancang untuk memulihkan ketersediaan instans saat AWS mendeteksi masalah perangkat keras atau perangkat lunak yang mendasari yang menyebabkan pemeriksaan status sistem gagal. Jika saat ini Anda mengalami masalah saat mengakses instans, lihat Memecahkan masalah inst ans EC2.

Jika AWS mendeteksi bahwa instance tidak tersedia karena masalah perangkat keras atau perangkat lunak yang mendasarinya, pemulihan berbasis CloudWatch tindakan dapat secara otomatis memulihkan ketersediaan instans dengan memindahkan instance dari host dengan masalah mendasarinya ke host lain.

Jika pemulihan berbasis CloudWatch tindakan terjadi, AWS kirim salah satu peristiwa berikut ke Anda Dasbor AWS Health, tergantung pada hasilnya:

- Acara sukses: `AWS_EC2_INSTANCE_AUTO_RECOVERY_SUCCESS`
- Peristiwa kegagalan: `AWS_EC2_INSTANCE_AUTO_RECOVERY_FAILURE`

Anda dapat mengonfigurasi pemulihan berbasis CloudWatch tindakan untuk menambahkan tindakan pemulihan ke CloudWatch alarm Amazon. CloudWatch pemulihan berbasis tindakan bekerja dengan met `StatusCheckFailed_System` rik. CloudWatchpemulihan berbasis tindakan memberikan perincian waktu respons pemulihan hingga menit dan pemberitahuan Amazon Simple Notification Service (Amazon SNS) tentang tindakan dan hasil pemulihan. Opsi konfigurasi ini memungkinkan upaya pemulihan yang lebih cepat dengan kontrol yang lebih terperinci atas respons peristiwa kegagalan pemeriksaan status sistem dibandingkan dengan pemulihan otomatis yang disederhanakan. Untuk informasi selengkapnya tentang CloudWatch opsi yang tersedia, lihat Pemer iksaan status untuk instans Anda.

Namun, pemulihan berbasis CloudWatch tindakan hanya dapat beroperasi jika instance berada dalam `running` status, tidak ada peristiwa layanan yang tercantum di Dasbor AWS Health, dan ada kapasitas yang tersedia untuk jenis instans. Dalam beberapa situasi, seperti pemadaman yang signifikan, kendala kapasitas dapat menyebabkan upaya pemulihan gagal. Untuk informasi selengkapnya, lihat the section called “Pemecahan Masalah”.

> **Warning**
>
> Saat AWS memulihkan instance Anda karena masalah perangkat keras atau perangkat lunak yang mendasarinya, perhatikan konsekuensi berikut: data yang disimpan dalam memori volatile (RAM) dan volume penyimpanan instans akan hilang, dan waktu aktif sistem operasi akan dimulai dari nol. Untuk membantu melindungi dari kehilangan data, kami menyarankan Anda membuat cadangan data berharga secara teratur. Untuk informasi selengkapnya tentang praktik terbaik pencadangan dan pemulihan untuk instans EC2, lihat Prakti k terbaik untuk Amazon EC 2.
>
> Mekanisme pemulihan instans otomatis dirancang untuk instance individual. Untuk panduan membangun sistem yang tangguh, lihatMembangun sistem yang tangguh.

Daftar Isi


- Persyaratan untuk mengaktifkan pemulihan berbasis CloudWatch tindakan
- Konfigur CloudWatch asikan pemulihan berbasis tindakan
- Memecahkan masalah kegagalan pemulihan berbasis CloudWatch tindakan

## Persyaratan untuk mengaktifkan pemulihan berbasis CloudWatch tindakan

CloudWatch pemulihan berbasis tindakan dapat diaktifkan pada instans yang memenuhi kriteria berikut:

Tipe instans

- Tujuan umum: A1, M3, M4, M5, M5a, M5n, M5zn, M6a, M6g, M6i, M6in, M7a, M7g, M7i,, M8a, M8azN, M8g, M8g, M8gn, M8i,, M8in, M8iB M7i-flex, M9g, T1, T2, T3, T3a, T4g M8i-flex
- Komputasi dioptimalkan: C 3, C4, C5, C5a, C5n, C6a, C6g, C6gn, C6i, C6in, C7a, C7g, C7gn, C7i,, C8a, C8g, C8Gn, C8i,, C8in, C8iN, C8iB, C9g C7i-flex C8i-flex
- Memori dioptimalkan: R3, R4, R5, R5a, R5b, R5n, R6a, R6g, R6i, R6in, R7a, R7g, R7i, R7iz, R8a, R8g, R8g, R8g, R8i,, R8in, R8ib, R9g,,,,,,,,,, X1, X1e, x2IDn, x2IEDn, X2IEZn, X8g R8i-flex, X8i U-3tb1 U-6tb1 U-9tb1 U-12tb1 U-18tb1 U-24tb1 U7i-6tb U7i-8tb U7i-12tb U7in-16tb U7in-24tb U7in-32tb U7inh-32tb
- Komputasi yang dipercepat: G3, G5g, Inf1, P3, VT1
- High-performance komputasi: hpc6a, hpc7a, hpc7g, hpc8a
- Instans logam: Salah satu jenis instance di atas dengan ukuran instance logam.
- Jika volume penyimpanan instance ditambahkan saat peluncuran: Maka hanya jenis instance berikut yang didukung: M3, C3, R3, X1, X1e, X2idn, X2iEDN

Penghunian

- Bersama
- Instans Khusus

  Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2.

Batasan

CloudWatch pemulihan berbasis tindakan tidak didukung untuk contoh dengan karakteristik berikut:

- Penyewaan: Tuan Rumah Khusus. Untuk Host Khusus, gunakan Pemulihan Otom atis Host Khusus sebagai gantinya.
- Jaringan: Instans menggunakan Adaptor Kain Elastis
- Penskalaan Otomatis: Instans yang merupakan bagian dari grup Penskalaan Otomatis
- Pemeliharaan: Instans saat ini sedang menjalani acara pemeliharaan terjadwal

### Menemukan jenis instans yang didukung

Anda dapat melihat jenis instans yang mendukung pemulihan berbasis CloudWatch tindakan.

#### Console

Untuk melihat jenis instans yang mendukung pemulihan berbasis CloudWatch tindakan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Tipe Instans.
3. Di bilah filter, tambahkan filter Auto Recovery support = true. Tabel jenis Instance menampilkan semua jenis instans yang mendukung pemulihan berbasis CloudWatch tindakan.
4. (Opsional) Tambahkan filter ke cakupan lebih lanjut ke jenis instans tertentu yang menarik.

#### AWS CLI

Untuk melihat jenis instans yang mendukung pemulihan berbasis CloudWatch tindakan

Gunakan perintah describe-instance-types dengan filter. `auto-recovery-supported`

```
aws ec2 describe-instance-types \
    --filters Name=auto-recovery-supported,Values=true \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

#### PowerShell

Untuk melihat jenis instans yang mendukung pemulihan berbasis CloudWatch tindakan

Gunakan Get-EC2InstanceType cmdlet dengan filter. `auto-recovery-supported`

```
Get-EC2InstanceType `
    -Filter @{Name="auto-recovery-supported";Values="true"} | `
    Select InstanceType | Sort-Object InstanceType
```

## Konfigur CloudWatch asikan pemulihan berbasis tindakan

Untuk mengonfigurasi pemulihan berbasis CloudWatch tindakan untuk instans EC2, buat CloudWatch alarm yang memantau `StatusCheckFailed_System` metrik untuk instans yang

ditentukan. Atur alarm untuk memicu ketika nilai metrik adalah 1, menunjukkan pemeriksaan status sistem yang gagal. Konfigurasikan tindakan alarm untuk memulihkan instance secara otomatis saat dipicu.

Anda dapat mengonfigurasi alarm menggunakan konsol Amazon EC2 atau CloudWatch konsol. Untuk petunjuknya, lihat Tambahkan tindakan pemulihan ke CloudWatch alarm Amazon di panduan pengguna ini, atau Men ambahkan tindakan pemulihan ke CloudWatch alarm Amazon di Panduan CloudWatch Pengguna Amazon.

### Memecahkan masalah kegagalan pemulihan berbasis CloudWatch tindakan

Jika pemulihan berbasis CloudWatch tindakan gagal memulihkan instans Anda, pertimbangkan masalah berikut:

- AWS acara layanan sedang berjalan

  CloudWatch pemulihan berbasis tindakan tidak beroperasi selama peristiwa layanan di Dasbor AWS Health. Anda mungkin tidak menerima notifikasi kegagalan pemulihan untuk peristiwa semacam itu. Untuk informasi ketersediaan layanan terbaru, lihat halaman Status kesehatan layanan.
- Kapasitas tidak mencukupi

  Perangkat keras pengganti sementara tidak mencukupi untuk memigrasikan instance.
- Upaya pemulihan harian maksimum tercapai

  Instance telah mencapai tunjangan harian maksimum untuk upaya pemulihan. Instans Anda mungkin akan dihentikan jika pemulihan otomatis gagal dan degradasi perangkat keras ditentukan sebagai akar penyebab pemeriksaan status sistem yang gagal.

Jika kegagalan pemeriksaan status sistem instans tetap ada meskipun ada beberapa upaya pemulihan, lihat Memec ahkan masalah instans dengan pemeriksaan status gagal untuk panduan tambahan.

## Menggunakan metadata instans untuk mengelola instans EC2 Anda

Metadata instans adalah data tentang instans Anda yang dapat Anda gunakan untuk mengonfigurasi atau mengelola instans berjalan. Metadata instans mencakup yang berikut:

Properti metadata instans

Properti metadata instans dibagi menjadi beberapa kategori, misalnya nama host, acara, dan grup keamanan.

Data dinamis

Data dinamis adalah metadata yang dihasilkan saat instance diluncurkan, seperti dokumen identitas instance. Untuk informasi selengkapnya, lihat Kategori data dinamis.

Data pengguna

Anda juga dapat menggunakan metadata instans untuk mengakses data pengguna yang Anda tentukan saat meluncurkan instans. Misalnya, Anda dapat menentukan parameter untuk mengonfigurasi instans Anda, atau menyertakan skrip sederhana. Anda juga dapat membuat AMI generik dan menggunakan data pengguna untuk memodifikasi file konfigurasi yang disediakan pada waktu peluncuran. Misalnya, jika Anda menjalankan server web untuk berbagai bisnis kecil, mereka semua dapat menggunakan AMI generik yang sama dan mengambil kontennya dari bucket Amazon S3 yang Anda tentukan dalam data pengguna saat peluncuran. Untuk menambahkan pelanggan baru kapan saja, buat bucket untuk pelanggan, tambahkan konten mereka, dan luncurkan AMI Anda dengan nama keranjang bucket unik yang diberikan untuk kode Anda di data pengguna. Jika Anda meluncurkan beberapa instans menggunakan `RunInstances` panggilan yang sama, data pengguna tersedia untuk semua instans dalam reservasi tersebut. Setiap instance yang merupakan bagian dari reservasi yang sama memiliki `ami-launch-index` nomor unik, sehingga Anda dapat menulis kode yang mengontrol apa yang dilakukan instance. Misalnya, host pertama mungkin memilih dirinya sendiri sebagai simpul asli dalam sebuah klaster. Untuk contoh peluncuran AMI mendetail, lihat .Identifikasi setiap instance yang diluncurkan dalam satu permintaan

**Important**

Meskipun Anda hanya dapat mengakses metadata instans dan data pengguna dari dalam instans itu sendiri, data tersebut tidak dilindungi oleh metode autentikasi atau kriptografi. Siapa pun yang memiliki akses langsung ke instans, dan perangkat lunak apa pun yang kemungkinan berjalan di instans, akan dapat melihat metadatanya. Oleh karena itu, Anda tidak boleh menyimpan data sensitif, seperti sandi atau kunci enkripsi yang tahan lama, sebagai data pengguna.

Daftar Isi


- Kategori metadata instans
- Kategori data dinamis
- Mengakses metadata instans untuk instans EC2
- Mengkonfigurasi opsi Layanan Metadata Instance
- Jalankan perintah saat Anda meluncurkan instance EC2 dengan input data pengguna
- Identifikasi setiap instance yang diluncurkan dalam satu permintaan


## Kategori metadata instans

Properti metadata instance dibagi menjadi beberapa kategori. Untuk mengambil properti metadata instance, Anda menentukan kategori dalam permintaan, dan metadata dikembalikan dalam respons.

Saat kategori baru dirilis, build metadata instans baru dibuat dengan nomor versi baru. Dalam tabel berikut, kolom Versi saat kategori dirilis menentukan versi build saat kategori metadata instans dirilis. Untuk menghindari keharusan memperbarui kode Anda setiap kali Amazon EC2 merilis build metadata instans baru, gunakan `latest`, bukan nomor versi, di permintaan metadata Anda. Untuk informasi selengkapnya, lihat Dapatkan versi metadata instans yang tersedia.

Saat Amazon EC2 merilis kategori metadata instans baru, metadata instans untuk kategori baru mungkin tidak tersedia untuk instans yang sudah ada. Dengan Nitro-based instance, Anda dapat mengambil metadata instance hanya untuk kategori yang tersedia saat peluncuran. Untuk instans dengan hypervisor Xen, Anda dapat menghentikan dan kemudian memulai instans untuk memperbarui kategori yang tersedia untuk instans tersebut.

Tabel berikut mencantumkan kategori metadata instans. Beberapa nama kategori menyertakan placeholder untuk data yang unik untuk instans Anda. Misalnya, *mac* mewakili alamat MAC untuk antarmuka jaringan. Anda harus mengganti placeholder dengan nilai sebenarnya saat Anda mengambil metadata instans.

| Kategori | Deskripsi | Versi ketika kategori dirilis |
|---|---|---|
| `ami-id` | ID AMI yang digunakan untuk meluncurkan instans. | 1.0 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `ami-launch-index` | Jika Anda meluncurkan beberapa instans menggunakan `RunInstances` panggilan yang sama, nilai ini menunjukkan urutan peluncuran untuk setiap instans. Nilai instans pertama yang diluncurkan adalah 0. Jika Anda meluncurkan instans menggunakan Auto Scaling atau armada EC2, nilai ini selalu 0. | 1.0 |
| `ami-manifest-path` | Jalur ke file manifes AMI di Amazon S3. Jika Anda menggunakan Amazon EBS-backed AMI untuk meluncurkan instance, hasil yang dikembalikan adalah`unknown`. | 1.0 |
| `ancestor-ami-ids` | ID AMI dari setiap instans yang dibundel ulang untuk membuat AMI ini. Nilai ini hanya akan ada jika file manifes AMI berisi kunci `ancestor-amis` . | 10/10/2007 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `autoscaling/target-lifecycle-state` | Nilai yang menunjukkan status siklus hidup Auto Scaling target yang dialihkan oleh instans Auto Scaling. Anda pada saat instans bertransisi ke salah satu status siklus hidup target setelah 10 Maret 2022. Nilai yang mungkin: `Detached` `InService` \| `Standby` \| `Terminated` \| `Warmed:Hibernated` \| `Warmed:Running` \| `Warmed:Stopped` \| `Warmed:Terminated`. Lihat Mengambil status siklus hidup target melalui metadata instans di Panduan Pengguna Amazon EC2 Auto Scaling. | 2021-07-15 |
| `block-device-mapping/ami` | Perangkat virtual yang berisi sistem root/boot file. | 15/12/2007 |
| `block-device-mapping/ebsN` | Perangkat virtual yang terkait dengan volume Amazon EBS apa pun. Volume Amazon EBS hanya tersedia dalam metadata jika volume itu ada pada waktu peluncuran atau saat instans terakhir kali dimulai. N menunjukkan indeks volume Amazon EBS (seperti `ebs1` atau `ebs2`). | 15/12/2007 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `block-device-mapping/ephemeral N` | Perangkat virtual untuk setiap volume penyimpanan instans non-NVMe. N menunjukkan indeks setiap volume. Jumlah volume penyimpanan instans dalam pemetaan perangkat blok mungkin tidak cocok dengan jumlah volume penyimpanan instans yang sebenarnya untuk instans tersebut. Tipe instans menentukan jumlah volume penyimpanan instans yang tersedia untuk sebuah instans. Jika jumlah volume penyimpanan instans dalam pemetaan perangkat blok melebihi jumlah yang tersedia untuk sebuah instans, volume penyimpanan instans tambahan akan diabaikan. | 15/12/2007 |
| `block-device-mapping/root` | Perangkat virtual atau partisi yang terkait dengan perangkat atau partisi root pada perangkat virtual, di mana sistem file root (/ atau C:) dikaitkan dengan instans tertentu. | 15/12/2007 |
| `block-device-mapping/swap` | Perangkat virtual yang terkait dengan `swap`. Tidak selalu ada. | 15/12/2007 |
| `events/maintenance/history` | Jika ada peristiwa pemeliharaan yang diselesaikan atau dibatalkan untuk instans tersebut, berisi string JSON dengan informasi tentang peristiwa tersebut. | 17/08/2018 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `events/maintenance/scheduled` | Jika ada peristiwa pemeliharaan aktif untuk instans tersebut, berisi string JSON dengan informasi tentang peristiwa tersebut. Untuk informasi selengkapnya, lihat Melihat acara terjadwal yang memengaruhi instans Amazon EC2 Anda. | 17/08/2018 |
| `events/recommendations/rebalance` | Perkiraan waktu, dalam UTC, ketika notifikasi rekomendasi penyeimbangan ulang instans EC2 dikeluarkan untuk instans tersebut. Berikut adalah contoh metadata untuk kategori ini: `{"noticeTime": "2020-11-05T08:22:00Z"}` . Kategori ini hanya tersedia setelah notifikasi dikeluarkan. Untuk informasi selengkapnya, lihat Rekomendasi penyeimbangan ulang instans EC2. | 2020-10-27 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `hostname` | Jika instance EC2 menggunakan IP-based penamaan (IPBN), ini adalah nama host DNS IPv4 pribadi dari instance tersebut. Jika instance EC2 menggunakan Resource-based penamaan (RBN), ini adalah RBN. Jika ada banyak antarmuka jaringan, ini mengacu pada perangkat eth0 (perangkat yang nomor perangkatnya adalah 0). Untuk informasi selengkapnya tentang IPBN dan RBN, lihat Nama host dan domain instans EC2. | 1.0 |
| `iam/info` | Jika ada peran IAM yang terkait dengan instance, berisi informasi tentang terakhir kali profil instans diperbarui, termasuk LastUpdated tanggal instans, InstanceProfileArn, danInstanceProfileId. Jika tidak, tidak ada. | 12/01/2012 |
| `iam/security-credentials/role-name` | Jika ada peran IAM yang terkait dengan instance, *`role-name`* adalah nama peran, dan *`role-name`* berisi kredenSIAL keamanan sementara yang terkait dengan peran tersebut (untuk informasi selengkapnya, lihatMengambil kredensial keamanan dari metadata instans). Jika tidak, tidak ada. | 12/01/2012 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `identity-credentials/ec2/info` | Informasi tentang kredensial di `identity-credentials/ec2/security-credentials/ec2-instance` . | 23/05/2018 |
| `identity-credentials/ec2/security-credentials/ec2-instance` | KredenSIAL untuk peran identitas instans yang memungkinkan perangkat lunak instans mengidentifikasi dirinya sendiri AWS untuk mendukung fitur seperti EC2 Instance Connect dan Def AWS Systems Manager ault Host Management Configuration. KredenSIAL ini tidak memiliki kebijakan yang dilampirkan, sehingga mereka tidak memiliki izin AWS API tambahan selain mengidentifikasi instance ke AWS fitur. Untuk informasi selengkap nya, lihat Peran identitas instans untuk instans Amazon EC2. | 23/05/2018 |
| `instance-action` | Memberi tahu instans bahwa instans harus di-boot ulang sebagai persiapan untuk pemaketan. Nilai yang valid: `none` \| `shutdown` \| `bundle-pending` . | 01/09/2008 |
| `instance-id` | ID instans ini. | 1.0 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `instance-life-cycle` | Opsi pembelian instans ini. Untuk informasi selengkapnya, lihat Opsi penagihan dan pembelian Amazon EC2. | 01/10/2019 |
| `instance-type` | Tipe instans. Untuk informasi selengkapnya, lihat Tipe instans Amazon EC2. | 29/08/2007 |
| `ipv6` | Alamat IPv6 instans. Jika ada banyak antarmuka jaringan, alamat ini mengacu pada antarmuka jaringan perangkat eth0 (perangkat dengan nomor perangkat 0) dan alamat IPv6 yang pertama ditetapkan. Jika tidak ada alamat IPv6 di antarmuka jaringan [0], item ini tidak disetel dan menghasilkan respons HTTP 404. | 2021-01-03 |
| `kernel-id` | ID kernel yang diluncurkan dengan instans ini, jika ada. | 01/02/2008 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `local-hostname` | Jika ada banyak antarmuka jaringan, ini mengacu pada perangkat eth0 (perangkat yang nomor perangkatnya adalah 0). Jika instance EC2 menggunak an IP-based penamaan (IPBN), ini adalah nama host DNS IPv4 pribadi dari instance tersebut. Jika instance EC2 menggunak an Resource-based penamaan (RBN), ini adalah RBN. Untuk informasi selengkapnya tentang penamaan IPBN, RBN, dan EC2, lihat Nama host dan domain instans EC2. | 19/01/2007 |
| `local-ipv4` | Alamat IPv4 privat dari instans. Jika ada banyak antarmuka jaringan, ini mengacu pada perangkat eth0 (perangkat yang nomor perangkatnya adalah 0). Jika ini adalah IPv6-only contoh, item ini tidak disetel dan menghasil kan respons HTTP 404. | 1.0 |
| `mac` | Alamat kontrol akses media (MAC) instans. Jika ada banyak antarmuka jaringan, ini mengacu pada perangkat eth0 (perangkat yang nomor perangkatnya adalah 0). | 01/01/2011 |
| `metrics/vhostmd` | Tidak lagi tersedia. | 01/05/2011 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `network/interfaces/macs/mac/device-number` | Nomor perangkat unik yang terkait dengan antarmuka itu. Nomor perangkat sesuai dengan nama perangkat; misalnya, `device-number` pada 2 adalah untuk perangkat eth2. Kategori ini sesuai dengan bidang `DeviceIndex` dan `device-index` yang digunakan oleh API Amazon EC2 dan perintah EC2 untuk AWS CLI. | 01-01-2011 |
| `network/interfaces/macs/mac/interface-id` | ID antarmuka jaringan. | 01-01-2011 |
| `network/interfaces/macs/mac/ipv4-associations/public-ip` | Alamat IPv4 yang terkait dengan setiap alamat IP publik dan ditetapkan ke antarmuka itu. | 01-01-2011 |
| `network/interfaces/macs/mac/ipv6s` | Alamat IPv6 ditetapkan ke antarmuka. | 30/06/2016 |
| `network/interfaces/macs/mac/ipv6-prefix` | Awalan IPv6 ditetapkan ke antarmuka jaringan. | |
| `network/interfaces/macs/mac/local-hostname` | Nama host DNS IPv4 pribadi dari antarmuka jaringan. Nilai ini tidak diatur untuk antarmuka tanpa alamat IP sama sekali, seperti EFA-only antarmuka. Jika ini adalah IPv6-only antarmuka, ini adalah nama berbasis sumber daya. Untuk informasi selengkapnya tentang IPBN dan RBN, lihat Nama host dan domain instans EC2. | 19/01/2007 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `network/interfaces/macs/mac/local-ipv4s` | Alamat IPv4 privat yang terkait dengan antarmuka. Jika ini adalah antarmuka IPv6-only jaringan, item ini tidak disetel dan menghasilkan respons HTTP 404. | 01-01-2011 |
| `network/interfaces/macs/mac/mac` | Alamat MAC instans. | 01/01/2011 |
| `network/interfaces/macs/`*`mac`*`/network-card` | Indeks kartu jaringan. Beberapa tipe instans mendukung banyak kartu jaringan. | 01/11/2020 |
| `network/interfaces/macs/mac/owner-id` | ID pemilik antarmuka jaringan. Di lingkungan multi-antarmuka, antarmuka dapat dipasang oleh pihak ketiga, seperti Elastic Load Balancing. Lalu lintas pada antarmuka selalu ditagihkan ke pemilik antarmuka. | 01/01/2011 |
| `network/interfaces/macs/mac/public-hostname` | DNS publik antarmuka (IPv4). Kategori ini hanya ditampilkan jika atribut `enableDnsHostnames` diatur ke `true`. Untuk informasi selengkapnya, lihat Atribut DNS untuk VPC Anda dalam Panduan Pengguna Amazon VPC. Jika instans hanya memiliki alamat IPv6 publik dan tidak ada alamat IPv4 publik, item ini tidak diatur dan akan menghasilkan respons HTTP 404. | 01-01-2011 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `network/interfaces/macs/mac/public-ipv4s` | Alamat IP publik atau alamat IP Elastis yang terkait dengan antarmuka. Mungkin ada beberapa alamat IPv4 pada sebuah instans. | 01-01-2011 |
| `network/interfaces/macs/mac/security-groups` | Grup keamanan yang memiliki antarmuka jaringan. | 01-01-2011 |
| `network/interfaces/macs/mac/security-group-ids` | ID grup keamanan yang memiliki antarmuka jaringan. | 01-01-2011 |
| `network/interfaces/macs/mac/subnet-id` | ID subnet tempat antarmuka berada. | 01-01-2011 |
| `network/interfaces/macs/mac/subnet-ipv4-cidr-block` | Blok CIDR IPv4 dari subnet tempat antarmuka berada. | 01-01-2011 |
| `network/interfaces/macs/mac/subnet-ipv6-cidr-blocks` | Blok CIDR IPv6 dari subnet tempat antarmuka berada. | 30/06/2016 |
| `network/interfaces/macs/mac/vpc-id` | ID VPC tempat antarmuka berada. | 01-01-2011 |
| `network/interfaces/macs/mac/vpc-ipv4-cidr-block` | Blok CIDR IPv4 primer dari VPC. | 01-01-2011 |
| `network/interfaces/macs/mac/vpc-ipv4-cidr-blocks` | Blok CIDR IPv4 untuk VPC. | 30/06/2016 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `network/interfaces/macs/mac/vpc-ipv6-cidr-blocks` | Blok CIDR IPv6 dari VPC tempat antarmuka berada. | 30/06/2016 |
| `placement/availability-zone` | Zona Ketersediaan tempat instans diluncurkan. | 01/02/2008 |
| `placement/availability-zone-id` | ID Zona Ketersediaan statis tempat instans diluncurkan. ID Zona Ketersediaan konsisten di semua akun. Namun, ini mungkin berbeda dari Zona Ketersediaan, yang dapat berbeda tergantung pada akun. | 01/10/2019 |
| `placement/group-name` | Nama grup penempatan tempat instans diluncurkan. | 24/08/2020 |
| `placement/host-id` | ID host tempat instans diluncurkan. Hanya berlaku untuk Host Khusus. | 24/08/2020 |
| `placement/partition-number` | Jumlah partisi tempat instans diluncurkan. | 24/08/2020 |
| `placement/region` | Wil AWS ayah tempat instance diluncurkan. | 24/08/2020 |
| `product-codes` | AWS Marketplace kode produk yang terkait dengan instance, jika ada. | 01/03/2007 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `public-hostname` | DNS publik instans (IPv4). Kategori ini hanya ditampilkan jika atribut `enableDnsHostnames` diatur ke `true`. Untuk informasi selengkapnya, lihat Atribut DNS untuk VPC Anda dalam Panduan Pengguna Amazon VPC. Jika instans hanya memiliki alamat IPv6 publik dan tidak ada alamat IPv4 publik, item ini tidak diatur dan akan menghasilkan respons HTTP 404. | 19/01/2007 |
| `public-ipv4` | Alamat IPv4 publik. Jika alamat IP Elastis dikaitkan dengan instans, nilai yang ditampilkan adalah alamat IP Elastis. | 19/01/2007 |
| `public-keys/0/openssh-key` | Kunci publik. Hanya tersedia jika disediakan pada waktu peluncuran instans. | 1.0 |
| `ramdisk-id` | ID dari RAM disk ditentukan pada waktu peluncuran, jika ada. | 10/10/2007 |
| `reservation-id` | ID reservasi. | 1.0 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `security-groups` | Nama-nama grup keamanan yang diterapkan ke instans.<br><br>Setelah peluncuran, Anda dapat mengubah grup keamanan instans. Perubahan tersebut tercermin di sini dan di network/interfaces /macs//security-groups. *`mac`* | 1.0 |
| `services/domain` | Domain untuk AWS sumber daya untuk Wilayah. | 25/02/2014 |
| `services/partition` | Partisi tempat sumber daya berada. Untuk Wil AWS ayah standar, partisi adalah`aws`. Jika Anda memiliki sumber daya di partisi lain, maka partisi-nya adalah `aws-`*`partitionname`* . Contohnya, partisi untuk sumber daya di Wilayah Tiongkok (Beijing) adalah `aws-cn`. | 20/10/2015 |
| `spot/instance-action` | Tindakan (hibernasi, berhenti, atau berakhir) dan perkiraan waktu, dalam UTC, saat tindakan akan terjadi. Item ini ada hanya jika Instans Spot telah ditandai untuk hibernasi, berhenti, atau berakhir. Untuk informasi selengkapnya, lihat instance-action. | 15/11/2016 |

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `spot/termination-time` | Perkiraan waktu, dalam UTC, sistem operasi untuk Instans Spot Anda akan menerima sinyal shutdown. Item ini hadir dan berisi nilai waktu (misalnya, 2015-01-05T18:02:00Z) hanya jika Instans Spot telah ditandai untuk penghentian oleh Amazon EC2. Item waktu pengakhiran tidak diatur ke suatu waktu jika Anda sendiri mengakhiri Instans Spot. Untuk informasi selengkapnya, lihat waktu penghentian. | 05/11/2014 |
| `system` | Jenis virtualisasi yang mendasari (hypervisor) dari instance. | 2022-09-24 |
| `tags/instance` | Tanda instans yang terkait dengan instans. Hanya tersedia jika Anda secara eksplisit mengizinkan akses ke tanda dalam metadata instans. Untuk informasi selengkapnya, lihat Aktifkan akses ke tag dalam metadata instance. | 2021-03-23 |

## Kategori data dinamis

Tabel berikut mencantumkan kategori data dinamis.

| Kategori | Deskripsi | Versi ketika kategori dirilis |
| --- | --- | --- |
| `fws/instance-monitoring` | Nilai yang menunjukkan apakah pelanggan telah mengaktifkan pemantauan satu menit yang terperinci. CloudWatch Nilai yang valid: `enabled \| disabled` | 04/04/2009 |
| `instance-identity/document` | JSON berisi atribut instance, seperti instance-id, alamat IP pribadi, dan banyak lagi. Lihat Dokumen identitas instans untuk instans Amazon EC2. | 04/04/2009 |
| `instance-identity/pkcs7` | Digunakan untuk memverifikasi keaslian dokumen dan konten terhadap tanda tangan. Lihat Dokumen identitas instans untuk instans Amazon EC2. | 04/04/2009 |
| `instance-identity/signature` | Data yang dapat digunakan pihak lain untuk memverifikasi asal dan keasliannya. Lihat Dokumen identitas instans untuk instans Amazon EC2. | 04/04/2009 |

## Mengakses metadata instans untuk instans EC2

Anda dapat mengakses metadata instans EC2 dari dalam instans itu sendiri atau dari konsol EC2, API, SDK, atau. AWS CLI Untuk mendapatkan pengaturan metadata instans saat ini untuk instance dari konsol atau baris perintah, lihatMengueri opsi metadata instans untuk instans yang ada.

Anda juga dapat memodifikasi data pengguna untuk instans dengan volume root EBS. Instans harus berada dalam status berhenti. Untuk petunjuk konsol, lihat Memperbarui data pengguna instance. Untuk contoh Linux yang menggunakan AWS CLI, lihat modif y-instance-atribut. Untuk contoh Windows yang menggunakan Tools untuk Windows PowerShell, lihatthe section called “Data pengguna dan Alat untuk Windows PowerShell”.

> **Note**
>
> Anda tidak dikenai biaya untuk permintaan HTTP yang digunakan untuk mengambil metadata instans dan data pengguna.

## Pertimbangan akses metadata instans

Untuk menghindari masalah dengan metadata instance, pertimbangkan hal berikut.

Kegagalan peluncuran instans karena penegakan IMDSv2 () **`HttpTokensEnforced=enabled`**

Sebelum mengaktifkan penegakan IMDSv2, Anda memerlukan semua perangkat lunak Anda pada instance untuk mendukung IMDSv2, setelah itu Anda dapat mengubah default untuk menonaktifkan IMDSv1 ()`httpTokens=required`, setelah itu Anda dapat mengaktifkan penegakan. Untuk informasi selengkapnya, lihat Transisi ke penggunaan Layanan Metadata Instans Versi 2.

Format perintah

Format perintah berbeda, tergantung pada apakah Anda menggunakan Instance Metadata Service Version 1 (IMDSv1) atau Instance Metadata Service Version 2 (IMDSv2). Secara default, Anda dapat menggunakan kedua versi Layanan Metadata Instance. Untuk mengharuskan penggunaan IMDSv2, lihat Menggunakan Layanan Metadata Instance untuk mengakses metadata instance.

Jika IMDSv2 diperlukan, IMDsv1 tidak berfungsi

Jika Anda menggunakan IMDSv1 dan tidak menerima tanggapan, kemungkinan IMDSv2 diperlukan. Untuk memeriksa apakah IMDSv2 diperlukan, pilih instance untuk melihat detailnya. Nilai IMDSv2 menunjukkan baik Diper lukan (Anda harus menggunakan IMDSv2) atau Opsional (Anda dapat menggunakan IMDSv2 atau IMDSv1).

(IMDSv2) Gunakan /latest/api/token untuk mengambil token

M PUT enerbitkan permintaan ke jalur khusus versi apa pun, misalnya`/2021-03-23/api/token`, menghasilkan layanan metadata mengembalikan 403 kesalahan Terlarang. Perilaku ini memang disengaja.

Versi metadata

Untuk menghindari keharusan memperbarui kode Anda setiap kali Amazon EC2 merilis build metadata instans baru, sebaiknya gunakan `latest` di jalur, dan bukan nomor versi.

Dukungan IPv6

Untuk mengambil metadata instans menggunakan alamat IPv6, pastikan Anda mengaktifkan dan menggunakan alamat IPv6 IMDS alih- `[fd00:ec2::254]` alih alamat IPv4. `169.254.169.254` Instance harus berupa Nitro-based instance yang diluncurkan di subnet yang mendukung IP v6.

(Windows) Buat AMI khusus menggunakan Windows Sysprep

Untuk memastikan bahwa IMDS berfungsi saat Anda meluncurkan instance dari AMI Windows kustom, AMI harus berupa gambar standar yang dibuat dengan Windows Sysprep. Jika tidak, IMDS tidak akan bekerja. Untuk informasi selengkapnya, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.

Dalam lingkungan kontainer, pertimbangkan konfigurasi ulang atau meningkatkan batas hop menjadi 2

AWS SDK menggunakan panggilan IMDSv2 secara default. Jika panggilan IMDSv2 tidak menerima respons, beberapa AWS SDK mencoba lagi panggilan dan, jika masih gagal, gunakan IMDSv1. Hal ini dapat mengakibatkan penundaan, terutama di lingkungan kontainer. Untuk AWS SDK yang memerlukan IMDSv2, jika batas hop adalah 1 di lingkungan kontainer, panggilan mungkin tidak menerima respons sama sekali karena pergi ke wadah dianggap sebagai hop jaringan tambahan.

Untuk mengurangi masalah ini di lingkungan kontainer, pertimbangkan untuk mengubah konfigurasi untuk meneruskan pengaturan (seperti Wilayah AWS) langsung ke wadah, atau pertimbangkan untuk meningkatkan batas hop menjadi 2. Untuk informasi tentang dampak batas hop, lihat Men ambahkan pertahanan secara mendalam terhadap firewall terbuka, proxy terbalik, dan kerentanan SSRF dengan penyempurnaan pada Layanan Metadata Instans EC2. Untuk informasi tentang mengubah batas hop, lihatMengubah batas hop respons PUT.

Batas paket per detik (PPS)

Ada batas 1024 paket per detik (PPS) untuk layanan yang menggunakan alamat link-local. Batas ini mencakup agregat permintaan Kueri DNS Resolver Route 53, permintaan Layanan Metadata Instance (IMDS), permintaan Amazon Time Service Network Time Protocol (NTP), dan permintaan Layanan Lisensi Windows (untuk instans berbasis Microsoft Windows).

Pertimbangan tambahan untuk akses data pengguna

- Data pengguna diperlakukan sebagai data buram: apa yang Anda tentukan adalah apa yang Anda dapatkan kembali setelah pengambilan. Terserah instans untuk menafsirkan dan bertindak berdasarkan data pengguna.
- Data pengguna harus dienkode base64. Tergantung pada alat atau SDK yang Anda gunakan, base64-encoding mungkin dilakukan untuk Anda. Contoh:
  - Konsol Amazon EC2 dapat mengenkode base64 untuk Anda atau menerima input yang dienkode base64.

  - AWS CLI versi 2 melakukan base64-encoding parameter biner untuk Anda secara default. AWS CLI versi 1 melakukan base64-encoding `--user-data` parameter untuk Anda.
  - Ini AWS SDK untuk Python (Boto3) melakukan base64-encoding `UserData` parameter untuk Anda.
- Data pengguna dibatasi hingga 16 KB, dalam bentuk mentah, sebelum dienkode base64. Ukuran string dengan panjang n setelah enkode base64 adalah ceil(n/3)*4.
- Data pengguna harus didekode base64 saat Anda mengambilnya. Jika Anda mengambil data menggunakan metadata instans atau konsol, data akan didekode untuk Anda secara otomatis.
- Jika Anda menghentikan sebuah instans, mengubah data penggunanya, dan memulai instans, data pengguna yang diperbarui tidak akan dijalankan secara otomatis saat Anda memulai instans. Dengan instans Windows, Anda dapat mengonfigurasi pengaturan sehingga skrip data pengguna yang diperbarui dijalankan satu kali saat Anda memulai instance atau setiap kali Anda me-reboot atau memulai instance.
- Data pengguna adalah atribut instans. Jika Anda membuat AMI dari instans, data pengguna instans tidak disertakan dalam AMI.

## Mengakses metadata instans dari dalam instans EC2

Karena metadata instans Anda tersedia dari instans berjalan, Anda tidak perlu menggunakan konsol Amazon EC2 atau AWS CLI. Hal ini berguna saat Anda menulis skrip yang akan dijalankan dari instans Anda. Misalnya, Anda dapat mengakses alamat IP lokal instans Anda dari metadata instans untuk mengelola koneksi ke aplikasi eksternal.

Semua hal berikut ini dianggap sebagai metadata instance, tetapi diakses dengan cara yang berbeda. Pilih tab yang mewakili jenis metadata instance yang ingin Anda akses untuk melihat informasi selengkapnya.

Metadata

Properti metadata instance dibagi menjadi beberapa kategori. Untuk deskripsi setiap kategori metadata instans, lihat Kategori metadata instans.

Untuk mengakses properti metadata instance dari dalam instance yang sedang berjalan, dapatkan data dari URI IPv4 atau IPv6 berikut. Alamat IP ini adalah alamat link-local dan hanya valid dari instance. Untuk informasi selengkapnya, lihat Link-local alamat.

IPv4

```
http://169.254.169.254/latest/meta-data/
```

IPv6

```
http://[fd00:ec2::254]/latest/meta-data/
```

Dynamic data

Untuk mengambil data dinamis dari dalam instance yang sedang berjalan, gunakan salah satu URI berikut.

IPv4

```
http://169.254.169.254/latest/dynamic/
```

IPv6

```
http://[fd00:ec2::254]/latest/dynamic/
```

Contoh: Akses dengan cURL

Contoh berikut digunakan cURL untuk mengambil kategori identitas instance tingkat tinggi.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/
rsa2048
pkcs7
document
signature
dsa2048
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/dynamic/instance-identity/
rsa2048
pkcs7
```

```
document
signature
dsa2048
```

Contoh: Akses dengan PowerShell

Contoh berikut digunakan PowerShell untuk mengambil kategori identitas instance tingkat tinggi.

IMDSv2

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/dynamic/instance-identity/
document
rsa2048
pkcs7
signature
```

IMDSv1

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/dynamic/instance-identity/
document
rsa2048
pkcs7
signature
```

Untuk informasi lebih lanjut tentang data dinamis dan contoh cara mengambilnya, lihat Dokumen identitas instans untuk instans Amazon EC2.

User data

Untuk mengambil data pengguna dari instance, gunakan salah satu URI berikut. Untuk mengambil data pengguna menggunakan alamat IPv6, Anda harus mengaktifkannya, dan instance harus berupa Nitro-based instance di subnet yang mendukung IPv6.

IPv4

```
http://169.254.169.254/latest/user-data
```

IPv6

```
http://[fd00:ec2::254]/latest/user-data
```

Permintaan untuk data pengguna mengembalikan data apa adanya (tipe konten `application/octet-stream`). Jika instans tidak memiliki data pengguna, permintaan akan mengembalikan `404 - Not Found`.

Contoh: Akses dengan cURL untuk mengambil teks yang dipisahkan koma

Contoh berikut digunakan cURL untuk mengambil data pengguna yang ditentukan sebagai teks yang dipisahkan koma.

IMDSv2

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

IMDSv1

```
curl http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

Contoh: Akses dengan PowerShell untuk mengambil teks yang dipisahkan koma

Contoh berikut digunakan PowerShell untuk mengambil data pengguna yang ditentukan sebagai teks yang dipisahkan koma.

IMDSv2

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

IMDSv1

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = Invoke-RestMethod -Headers
 @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
-Method PUT -Uri http://169.254.169.254/latest/api/token} -Method GET -uri
 http://169.254.169.254/latest/user-data
1234,john,reboot,true | 4512,richard, | 173,,,
```

Contoh: Akses dengan cURL untuk mengambil skrip

Contoh berikut digunakan cURL untuk mengambil data pengguna yang ditentukan sebagai skrip.

IMDSv2

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-
token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/user-
data
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

IMDSv1

```
curl http://169.254.169.254/latest/user-data
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

Contoh: Akses dengan PowerShell untuk mengambil skrip

Contoh berikut digunakan PowerShell untuk mengambil data pengguna yang ditentukan sebagai skrip.

IMDSv2

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds"
 = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri
 http://169.254.169.254/latest/user-data
```

```
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

IMDSv1

```
Invoke-RestMethod -uri http://169.254.169.254/latest/user-data
<powershell>
$file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

## Mengueri opsi metadata instans untuk instans yang ada

Anda dapat menanyakan opsi metadata instance untuk instans yang ada.

Console

Untuk menanyakan opsi metadata instance untuk instance yang ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance Anda dan periksa bidang berikut:
   - IMDSv2 — Nilai Wajib atau Opsional.
   - Izinkan tag dalam metadata instance — Nilainya di aktifkan atau Din onaktifkan.
4. Dengan instans Anda dipilih, pilih Tind akan, Peng aturan instans, Ubah opsi metadata instans.

   Kotak dialog menampilkan apakah layanan metadata instance diaktifkan atau dinonaktifkan untuk instance yang dipilih.

AWS CLI

Untuk menanyakan opsi metadata instance untuk instance yang ada

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-id i-1234567898abcdef0 \
    --query 'Reservations[].Instances[].MetadataOptions'
```

PowerShell

Untuk menanyakan opsi metadata instance untuk instance yang ada menggunakan Tools for PowerShell

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -InstanceId i-1234567898abcdef0).Instances.MetadataOptions
```

## Respons dan pesan kesalahan

Semua metadata instans ditampilkan sebagai teks (tipe konten HTTP `text/plain`).

Permintaan untuk sumber daya metadata tertentu mengembalikan nilai yang sesuai, atau kode kesalahan HTTP `404 - Not Found` jika sumber daya tidak tersedia.

Permintaan untuk sumber daya metadata umum (URI diakhiri dengan a /) mengembalikan daftar sumber daya yang tersedia, atau kode kesalahan HTTP `404 - Not Found` jika tidak ada sumber daya seperti itu. Item daftar berada di baris terpisah, diakhiri oleh feed baris (ASCII 10).

Jika permintaan IMDSv1 tidak menerima tanggapan, kemungkinan IMDSv2 diperlukan.

Untuk permintaan yang dibuat menggunakan IMDSv2, kode kesalahan HTTP berikut dapat dikembalikan:

- `400 - Missing or Invalid Parameters` – Permintaan PUT tidak valid.
- `401 - Unauthorized` – Permintaan GET menggunakan token yang tidak valid. Tindakan yang disarankan adalah membuat token baru.
- `403 - Forbidden` – Permintaan tidak diperbolehkan atau IMDS dimatikan.
- `404 - Not Found`Sumber daya tidak tersedia atau tidak ada sumber daya seperti itu.
- `503`— Permintaan tidak dapat diselesaikan. Coba lagi permintaannya.

Jika IMDS mengembalikan kesalahan, curl mencetak pesan kesalahan dalam output dan mengembalikan kode status sukses. Pesan kesalahan disimpan dalam `TOKEN` variabel, yang menyebabkan curl perintah yang menggunakan token gagal. Jika Anda memanggil curl dengan `-f` opsi, ia mengembalikan kode status kesalahan jika terjadi kesalahan server HTTP. Jika Anda mengaktifkan penanganan kesalahan, shell dapat menangkap kesalahan dan menghentikan skrip.

### Throttling kueri

Kami membatasi kueri ke IMDS per instans, dan kami membatasi jumlah koneksi simultan dari sebuah instans ke IMDS.

Jika Anda menggunakan IMDS untuk mengambil kredentif AWS keamanan, hindari meminta kredentif selama setiap transaksi atau secara bersamaan dari sejumlah besar utas atau proses, karena ini dapat menyebabkan pelambatan. Sebagai gantinya, kami menyarankan Anda menyimpan kredensial dalam cache hingga kredensial itu mendekati waktu kedaluwarsanya. Untuk informasi selengkapnya tentang peran IAM dan kredensial keamanan yang terkait dengan peran tersebut, lihat Mengambil kredensial keamanan dari metadata instans.

Jika Anda mengalami throttling saat mengakses IMDS, coba lagi kueri Anda dengan strategi mundur eksponensial.

### Menggunakan Layanan Metadata Instance untuk mengakses metadata instance

Anda dapat mengakses metadata instans dari instans yang sedang berjalan menggunakan salah satu metode berikut:

- Layanan Metadata Instans Versi 2 (IMDSv2) - metode berorientasi sesi

  Sebagai contoh, lihat Contoh untuk IMDSv2.
- Layanan Metadata Instance Versi 1 (IMDSv1) — sebuah metode request/response

  Sebagai contoh, lihat Contoh untuk IMDSv1.

Secara default, Anda dapat menggunakan IMDSv1 atau IMDSv2, atau keduanya.

Anda dapat mengonfigurasi Instance Metadata Service (IMDS) pada setiap instans untuk hanya menerima panggilan IMDSv2, yang akan menyebabkan panggilan IMDSv1 gagal. Untuk informasi tentang cara mengonfigurasi instans Anda agar menggunakan IMDSv2, lihat Mengkonfigurasi opsi Layanan Metadata Instance.

Header PUT atau GET bersifat unik untuk IMDSv2. Jika header ini ada dalam permintaan, maka permintaan tersebut ditujukan untuk IMDSv2. Jika header tidak ada, diasumsikan bahwa permintaan tersebut ditujukan untuk IMDSv1.

Untuk tinjauan ekstensif IMDSv2, lihat Tambahkan pertahanan secara mendalam terhadap firewall terbuka, proksi terbalik, dan kerentanan SSRF dengan peningkatan pada Layanan Metadata Instans EC2.

Topik


- Bagaimana cara kerja Layanan Metadata Instans Versi 2
- Gunakan yang didukung AWS SDK
- Contoh untuk IMDSv2
- Contoh untuk IMDSv1


## Bagaimana cara kerja Layanan Metadata Instans Versi 2

IMDSv2 menggunakan permintaan berorientasi sesi. Dengan permintaan berorientasi sesi, Anda membuat token sesi yang menentukan durasi sesi, yang bisa minimal satu detik dan maksimal enam jam. Selama durasi yang ditentukan, Anda dapat menggunakan token sesi yang sama untuk permintaan selanjutnya. Setelah durasi yang ditentukan berakhir, Anda harus membuat token sesi baru yang akan digunakan untuk permintaan di masa mendatang.

> **Note**
>
> Contoh di bagian ini menggunakan alamat IPv4 Layanan Metadata Instans (IMDS): `169.254.169.254`. Jika Anda mengambil metadata instans untuk instans EC2 melalui alamat IPv6, pastikan Anda mengaktifkan dan menggunakan alamat IPv6 sebagai gantinya: `[fd00:ec2::254]` Alamat IPv6 IMDS kompatibel dengan perintah IMDSv2. Alamat IPv6 hanya dapat diakses pada Nitro-based instance di IPv6-supported subnet (dual stack atau IPv6 saja).

Contoh berikut menggunakan skrip shell dan IMDSv2 untuk mengambil item metadata instance tingkat atas. Setiap contoh:

- Membuat token sesi yang berlangsung selama enam jam (21.600 detik) menggunakan permintaan PUT

- Menyimpan header token sesi dalam variabel bernama `TOKEN` (instance Linux) atau `token` (instans Windows)
- Meminta item metadata tingkat atas menggunakan token

Contoh Linux

Anda bisa menjalankan dua perintah terpisah, atau menggabungkannya.

Perintah terpisah

Pertama, hasilkan token menggunakan perintah berikut.

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"`
```

Kemudian, gunakan token untuk menghasilkan item metadata tingkat atas dengan menggunakan perintah berikut.

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
```

Perintah gabungan

Anda dapat menyimpan token dan menggabungkan perintah. Contoh berikut menggabungkan dua perintah di atas dan menyimpan header token sesi dalam variabel bernama TOKEN.

> **Note**
>
> Jika ada kesalahan dalam membuat token, alih-alih token yang valid, pesan kesalahan akan disimpan dalam variabel, dan perintah tidak akan bekerja.

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
 && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
```

Setelah Anda membuat token, Anda dapat menggunakannya kembali hingga kedaluwarsa. Dalam contoh perintah berikut, yang mendapatkan ID AMI yang digunakan untuk meluncurkan instans, token yang disimpan di `$TOKEN` dalam contoh sebelumnya digunakan kembali.

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/ami-id
```

Contoh Windows

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -
Uri http://169.254.169.254/latest/meta-data/
```

Setelah Anda membuat token, Anda dapat menggunakannya kembali hingga kedaluwarsa. Dalam contoh perintah berikut, yang mendapatkan ID AMI yang digunakan untuk meluncurkan instans, token yang disimpan di `$token` dalam contoh sebelumnya digunakan kembali.

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
 -Method GET -uri http://169.254.169.254/latest/meta-data/ami-id
```

Jika Anda menggunakan IMDSv2 untuk meminta metadata instans, maka permintaan tersebut harus menyertakan yang berikut ini:

1. Gunakan permintaan PUT untuk memulai sesi ke layanan metadata instans. Permintaan PUT mengembalikan sebuah token yang harus disertakan dalam permintaan GET selanjutnya ke layanan metadata instans. Token diperlukan untuk mengakses metadata menggunakan IMDSv2.
2. Sertakan token di semua permintaan GET ke IMDS. Saat penggunaan token diatur ke `required`, permintaan tanpa token yang valid atau dengan token yang kedaluwarsa akan menerima kode kesalahan HTTP `401 - Unauthorized`.
   - Token adalah kunci untuk instans tertentu. Token tidak valid di instans EC2 lainnya dan akan ditolak jika Anda mencoba menggunakannya di luar instans tempatnya dibuat.
   - Permintaan PUT harus menyertakan header yang menentukan waktu hidup (TTL) untuk token, dalam detik, hingga maksimum enam jam (21.600 detik). Token tersebut mewakili sesi logis. TTL menentukan lamanya waktu token itu valid dan, oleh karena itu, merupakan durasi sesi.
   - Setelah token kedaluwarsa, untuk terus mengakses metadata instans, Anda harus membuat sesi baru menggunakan PUT yang lain.
   - Anda dapat memilih untuk menggunakan kembali token atau membuat token baru dengan setiap permintaan. Untuk sejumlah kecil permintaan, mungkin lebih mudah untuk membuat dan

langsung menggunakan token setiap kali Anda perlu mengakses IMDS. Namun, untuk efisiensi, Anda dapat menentukan durasi yang lebih lama untuk token dan menggunakannya kembali daripada harus menulis perimtaan PUT setiap kali Anda perlu meminta metadata instans. Tidak ada batasan praktis untuk jumlah token yang bersamaan, masing-masing mewakili sesinya sendiri. Namun, IMDSv2 masih dibatasi oleh koneksi IMDS normal dan batas throttling. Untuk informasi selengkapnya, lihat Throttling kueri.

HTTP GET dan HEAD metode diperbolehkan dalam permintaan metadata instans IMDSv2. PUTpermintaan ditolak jika berisi X-Forwarded-For header.

Secara default, respons untuk permintaan PUT memiliki batas hop respons (waktu hidup) sebesar 1 di tingkat protokol IP. Jika Anda membutuhkan batas hop yang lebih besar, Anda dapat menyesuaikannya dengan menggunakan perintah modify- AWS CLI instance-metadata-options. Misalnya, Anda mungkin memerlukan batas hop yang lebih besar untuk kompatibilitas mundur dengan layanan container yang berjalan pada instans. Untuk informasi selengkapnya, lihat Mengonfigurasi opsi metadata instans untuk instans yang ada.

Gunakan yang didukung AWS SDK

Untuk menggunakan IMDSv2, instans EC2 Anda harus menggunakan versi AWS SDK yang mendukung penggunaan IMDSv2. Versi terbaru dari semua AWS SDK mendukung menggunakan IMDSv2.

> **Important**
>
> Kami menyarankan Anda untuk tetap mengikuti kabar terbaru terkait perilisan SDK untuk mendapatkan fitur, pembaruan keamanan, dan dependensi dasar terbaru. Penggunaan berkelanjutan dari versi SDK yang tidak didukung tidak disarankan dan dilakukan sesuai kebijaksanaan Anda. Untuk informasi selengkapnya, lihat kebijakan pemeliharaan SDK dan Alat AWS di Panduan Referensi SDK dan Alat AWS .

Berikut ini adalah versi minimum yang mendukung penggunaan IMDSv2:

- AWS CLI – 1.16.289
- AWS Tools for Windows PowerShell – 4.0.1.0
- AWS SDK untuk .NET – 3.3.634.1

- AWS SDK untuk C++ – 1.7.229
- AWS SDK untuk Go – 1.25.38
- AWS SDK untuk Go v2 - 0.19.0
- AWS SDK untuk Java – 1.11.678
- AWS SDK for Java 2.x – 2.10.21
- AWS SDK untuk JavaScript di Node.js - 2.722.0
- AWS SDK untuk Kotlin-- 1.1.4
- AWS SDK untuk PHP – 3.147.7
- AWS SDK untuk Python (Botocore) - 1.13.25
- AWS SDK untuk Python (Boto3) – 1.12.6
- AWS SDK untuk Ruby – 3.79.0

Contoh untuk IMDSv2

Jalankan contoh berikut pada instans Amazon EC2 Anda untuk mengambil metadata instans untuk IMDSv2.

Pada instance Windows, Anda dapat menggunakan Windows PowerShell atau Anda dapat menginstal cURL atau wget. Jika Anda menginstal alat pihak ketiga pada instance Windows, pastikan Anda membaca dokumentasi yang menyertainya dengan cermat, karena panggilan dan output mungkin berbeda dari apa yang dijelaskan di sini.

Contoh

- Dapatkan versi metadata instans yang tersedia
- Dapatkan item metadata tingkat atas
- Dapatkan nilai untuk item metadata
- Dapatkan daftar kunci publik yang tersedia
- Tunjukkan format di mana kunci publik 0 tersedia
- Dapatkan kunci publik 0 (dalam format kunci OpenSSH)
- Dapatkan ID subnet untuk instans
- Dapatkan tanda instans untuk sebuah instans

Dapatkan versi metadata instans yang tersedia

Contoh ini mendapatkan versi metadata instans yang tersedia. Setiap versi mengacu pada build metadata instans jika kategori metadata instans baru dirilis. Versi build metadata instans tidak berkorelasi dengan versi API Amazon EC2. Versi sebelumnya tersedia untuk Anda jika Anda memiliki skrip yang mengandalkan struktur dan informasi yang ada di versi sebelumnya.

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
```

```
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

Dapatkan item metadata tingkat atas

Contoh ini mendapatkan item metadata tingkat atas. Untuk informasi selengkapnya tentang item dalam respons, lihatKategori metadata instans.

Perhatikan bahwa tag disertakan dalam output ini hanya jika Anda telah mengizinkan akses. Untuk informasi selengkapnya, lihat the section called “Aktifkan akses ke tag dalam metadata instance”.

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hostname
iam/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
```

```
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
tags/
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
hostname
iam/
instance-action
instance-id
instance-life-cycle
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
```

```
tags/
```

Dapatkan nilai untuk item metadata

Contoh-contoh ini mendapatkan nilai dari beberapa item metadata tingkat atas yang diperoleh dalam contoh sebelumnya. Permintaan ini menggunakan token tersimpan yang dibuat menggunakan perintah pada contoh sebelumnya. Token tidak boleh kedaluwarsa.

cURL

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/reservation-id
r-0efghijk987654321
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
[ec2-user ~]$ curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/
latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

PowerShell

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/reservation-id
r-0efghijk987654321
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/local-hostname
```

```
ip-10-251-50-12.ec2.internal
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

Dapatkan daftar kunci publik yang tersedia

Contoh ini mendapatkan daftar kunci publik yang tersedia.

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/
0=my-public-key
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-keys/
0=my-public-key
```

Tunjukkan format di mana kunci publik 0 tersedia

Contoh ini menunjukkan format di mana kunci publik 0 tersedia.

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/
openssh-key
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
openssh-key
```

Dapatkan kunci publik 0 (dalam format kunci OpenSSH)

Contoh ini mendapatkan kunci publik 0 (di format kunci OpenSSH).

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method
 GET -Uri http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

```
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

Dapatkan ID subnet untuk instans

Contoh ini mendapatkan ID subnet untuk sebuah instans.

cURL

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/network/interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

PowerShell

```
PS C:\> [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
PS C:\> Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -
Method GET -Uri http://169.254.169.254/latest/meta-data/network/interfaces/
macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

Dapatkan tanda instans untuk sebuah instans

Jika akses ke tag instance dalam metadata instans diaktifkan, Anda bisa mendapatkan tag untuk instance dari metadata instance. Untuk informasi selengkapnya, lihat Mengambil tanda dari metadata instans.

Contoh untuk IMDSv1

Jalankan contoh berikut pada instans Amazon EC2 Anda untuk mengambil metadata instans untuk IMDSv1.

Pada instance Windows, Anda dapat menggunakan Windows PowerShell atau Anda dapat menginstal cURL atau wget. Jika Anda menginstal alat pihak ketiga pada instance Windows, pastikan Anda membaca dokumentasi yang menyertainya dengan cermat, karena panggilan dan output mungkin berbeda dari apa yang dijelaskan di sini.

Contoh

- Dapatkan versi metadata instans yang tersedia
- Dapatkan item metadata tingkat atas
- Dapatkan nilai untuk item metadata
- Dapatkan daftar kunci publik yang tersedia
- Tunjukkan format di mana kunci publik 0 tersedia
- Dapatkan kunci publik 0 (dalam format kunci OpenSSH)
- Dapatkan ID subnet untuk instans
- Dapatkan tanda instans untuk sebuah instans

Dapatkan versi metadata instans yang tersedia

Contoh ini mendapatkan versi metadata instans yang tersedia. Setiap versi mengacu pada build metadata instans jika kategori metadata instans baru dirilis. Versi build metadata instans tidak berkorelasi dengan versi API Amazon EC2. Versi sebelumnya tersedia untuk Anda jika Anda memiliki skrip yang mengandalkan struktur dan informasi yang ada di versi sebelumnya.

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/
```

```
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/
1.0
2007-01-19
2007-03-01
2007-08-29
2007-10-10
2007-12-15
2008-02-01
2008-09-01
2009-04-04
2011-01-01
2011-05-01
2012-01-12
2014-02-25
2014-11-05
2015-10-20
2016-04-19
...
latest
```

Dapatkan item metadata tingkat atas

Contoh ini mendapatkan item metadata tingkat atas. Untuk informasi selengkapnya tentang item dalam respons, lihatKategori metadata instans.

Perhatikan bahwa tag disertakan dalam output ini hanya jika Anda telah mengizinkan akses. Untuk informasi selengkapnya, lihat the section called “Aktifkan akses ke tag dalam metadata instance”.

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hostname
iam/
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
tags/
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
```

```
hostname
iam/
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
services/
tags/
```

Dapatkan nilai untuk item metadata

Contoh-contoh ini mendapatkan nilai dari beberapa item metadata tingkat atas yang diperoleh dalam contoh sebelumnya.

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/reservation-id
r-0efghijk987654321
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/ami-id
ami-0abcdef1234567890
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/reservation-id
r-0efghijk987654321
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/local-hostname
ip-10-251-50-12.ec2.internal
```

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-hostname
ec2-203-0-113-25.compute-1.amazonaws.com
```

Dapatkan daftar kunci publik yang tersedia

Contoh ini mendapatkan daftar kunci publik yang tersedia.

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/
0=my-public-key
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-keys/ 0=my-public-key
```

Tunjukkan format di mana kunci publik 0 tersedia

Contoh ini menunjukkan format di mana kunci publik 0 tersedia.

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/
```

```
openssh-key
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
keys/0/openssh-key
openssh-key
```

Dapatkan kunci publik 0 (dalam format kunci OpenSSH)

Contoh ini mendapatkan kunci publik 0 (di format kunci OpenSSH).

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/public-
keys/0/openssh-key
ssh-rsa MIICiTCCAfICCQD6m7oRw0uXOjANBgkqhkiG9w0BAQUFADCBiDELMAkGA1UEBhMC
VVMxCzAJBgNVBAgTAldBMRAwDgYDVQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6
b24xFDASBgNVBAsTC0lBTSBDb25zb2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAd
BgkqhkiG9w0BCQEWEG5vb25lQGFtYXpvbi5jb20wHhcNMTEwNDI1MjA0NTIxWhcN
MTIwNDI0MjA0NTIxWjCBiDELMAkGA1UEBhMCVVMxCzAJBgNVBAgTAldBMRAwDgYD
VQQHEwdTZWF0dGxlMQ8wDQYDVQQKEwZBbWF6b24xFDASBgNVBAsTC0lBTSBDb25z
b2xlMRIwEAYDVQQDEwlUZXN0Q2lsYWMxHzAdBgkqhkiG9w0BCQEWEG5vb25lQGFt
```

```
YXpvbi5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAMaK0dn+a4GmWIWJ
21uUSfwfEvySWtC2XADZ4nB+BLYgVIk60CpiwsZ3G93vUEIO3IyNoH/f0wYK8m9T
rDHudUZg3qX4waLG5M43q7Wgc/MbQITxOUSQv7c7ugFFDzQGBzZswY6786m86gpE
Ibb3OhjZnzcvQAaRHhdlQWIMm2nrAgMBAAEwDQYJKoZIhvcNAQEFBQADgYEAtCu4
nUhVVxYUntneD9+h8Mg9q6q+auNKyExzyLwaxlAoo7TJHidbtS4J5iNmZgXL0Fkb
FFBjvSfpJIlJ00zbhNYS5f6GuoEDmFJl0ZxBHjJnyp378OD8uTs7fLvjx79LjSTb
NYiytVbZPQUQ5Yaxu2jXnimvw3rrszlaEXAMPLE my-public-key
```

Dapatkan ID subnet untuk instans

Contoh ini mendapatkan ID subnet untuk sebuah instans.

cURL

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/network/interfaces/
macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

PowerShell

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/network/
interfaces/macs/02:29:96:8f:6a:2d/subnet-id
subnet-be9b61d7
```

Dapatkan tanda instans untuk sebuah instans

Jika akses ke tag instance dalam metadata instans diaktifkan, Anda bisa mendapatkan tag untuk instance dari metadata instance. Untuk informasi selengkapnya, lihat Mengambil tanda dari metadata instans.

## Transisi ke penggunaan Layanan Metadata Instans Versi 2

Jika Anda ingin mengonfigurasi instans agar hanya menerima panggilan Instance Metadata Service Version 2 (IMDSv2), sebaiknya gunakan alat berikut dan jalur transisi.

Topik

- Alat untuk transisi ke IMDSv2
- Jalur yang disarankan untuk mengharuskan IMDSv2

Alat untuk transisi ke IMDSv2

Alat berikut dapat membantu Anda mengidentifikasi, memantau, dan mengelola transisi perangkat lunak Anda dari IMDSv1 ke IMDSv2. Untuk petunjuk tentang cara menggunakan alat ini, lihatJalur yang disarankan untuk mengharuskan IMDSv2.

AWS perangkat lunak

Versi terbaru dari AWS CLI dan AWS SDK mendukung IMDSv2. Untuk menggunakan IMDSv2, perbarui instans EC2 Anda untuk menggunakan versi terbaru. Untuk versi AWS SDK minimum yang mendukung IMDSv2, lihat. Gunakan yang didukung AWS SDK

Semua paket perangkat lunak Amazon Linux 2 dan Amazon Linux 2023 mendukung IMDSv2. Amazon Linux 2023 menonaktifkan IMDSv1 secara default.

IMDS Package Analyzer

IMDS Packet Analyzer adalah alat sumber terbuka yang mengidentifikasi dan mencatat panggilan IMDSv1 selama fase boot instans dan operasi runtime Anda. Dengan menganalisis log ini, Anda dapat secara tepat mengidentifikasi perangkat lunak yang membuat panggilan IMDSv1 pada instance Anda dan menentukan apa yang perlu diperbarui untuk mendukung IMDSv2 hanya pada instans Anda. Anda dapat menjalankan IMDS Packet Analyzer dari baris perintah atau menginstalnya sebagai layanan. Untuk informasi lebih lanjut, lihat AWS ImdsPacketAnalyzer di GitHub.

CloudWatch

CloudWatch menyediakan dua metrik berikut untuk memantau instans Anda:

`MetadataNoToken`— IMDSv2 menggunakan sesi yang didukung token, sedangkan IMDSv1 tidak. Met `MetadataNoToken` rik melacak jumlah panggilan ke Instance Metadata Service (IMDS) yang menggunakan IMDSv1. Dengan melacak metrik ini ke nol, Anda dapat menentukan apakah dan kapan semua perangkat lunak Anda telah dimutakhirkan untuk menggunakan IMDSv2.

`MetadataNoTokenRejected`— Setelah menonaktifkan IMDSv1, Anda dapat menggunakan `MetadataNoTokenRejected` metrik untuk melacak berapa kali panggilan IMDSv1 dicoba dan ditolak. Dengan melacak metrik ini, Anda dapat memastikan apakah perangkat lunak Anda perlu diperbarui untuk menggunakan IMDSv2.

Untuk setiap instans EC2, metrik ini saling eksklusif. Ketika IMDSv1 diaktifkan (`httpTokens = optional`), hanya memancarkan. `MetadataNoToken` Ketika IMDSv1 dinonaktifkan

(`httpTokens = required`), hanya memancarkan. `MetadataNoTokenRejected` Untuk mengetahui kapan menggunakan metrik ini, lihatJalur yang disarankan untuk mengharuskan IMDSv2.

Untuk informasi selengkapnya, lihat Metrik instans.

Luncurkan API

Instans baru: Gunakan RunInstances API untuk meluncurkan instans baru yang memerlukan penggunaan IMDSv2. Untuk informasi selengkapnya, lihat Mengonfigurasi opsi metadata instans untuk instans baru.

Instans yang ada: Gunakan ModifyInstanceMetadataOptions API untuk mewajibkan penggunaan IMDSv2 pada instans yang ada. Untuk informasi selengkapnya, lihat Mengonfigurasi opsi metadata instans untuk instans yang ada.

Instans baru yang diluncurkan oleh grup Penskalaan Otom atis: Untuk mewajibkan penggunaan IMDSv2 pada semua instans baru yang diluncurkan oleh grup Penskalaan Otomatis, grup Penskalaan Otomatis Anda dapat menggunakan template peluncuran atau konfigurasi peluncuran. Saat Anda membuat templat peluncuran atau membuat konfigurasi peluncuran, Anda harus mengonfigurasi parameter `MetadataOptions` untuk mengharuskan penggunaan IMDSv2. Grup Auto Scaling meluncurkan instans baru menggunakan templat peluncuran atau konfigurasi peluncuran baru, tetapi instans yang ada tidak terpengaruh.

Instans yang ada dalam grup Penskalaan Otomatis: Gunakan ModifyInstanceMetadataOptions API untuk mewajibkan penggunaan IMDSv2 pada instans yang ada, atau hentikan instans dan grup Penskalaan Otomatis akan meluncurkan instans pengganti baru dengan pengaturan opsi metadata instance yang ditentukan dalam template peluncuran baru atau konfigurasi peluncuran.

AMI

AMI yang dikonfigurasi dengan `ImdsSupport` parameter yang disetel ke `v2.0` akan meluncurkan instans yang memerlukan IMDSv2 secara default. Amazon Linux 2023 dikonfigurasi dengan`ImdsSupport = v2.0`.

AMI baru: Gunakan perintah register-image CLI untuk mengatur `ImdsSupport` parameter ke `v2.0` saat membuat AMI baru.

AMI yang ada: Gunakan perintah CLI modify-image-atribut untuk mengatur `ImdsSupport` parameter saat memodifikasi AMI yang ada. `v2.0`

Untuk informasi selengkapnya, lihat Konfigurasikan AMI.

Account-level kontrol

Anda dapat mengonfigurasi nilai default untuk semua opsi metadata instans di tingkat akun. Nilai default diterapkan secara otomatis saat Anda meluncurkan instance. Untuk informasi lebih lanjut. lihatSetel IMDSv2 sebagai default untuk akun.

Anda juga dapat menegakkan persyaratan untuk menggunakan IMDSv2 di tingkat akun. Saat penegakan IMDSv2 diaktifkan:

- Instans baru: Instans yang dikonfigurasi untuk diluncurkan dengan IMDSv1 diaktifkan akan gagal diluncurkan
- Instans yang ada dengan IMDSv1 dinonaktifkan: Upaya untuk mengaktifkan IMDSv1 pada instans yang ada akan dicegah.
- Instans yang ada dengan IMDSv1 diaktifkan: Instans yang ada dengan IMDSv1 sudah diaktifkan tidak akan terpengaruh.

Untuk informasi selengkapnya, lihat Menerapkan IMDSv2 di tingkat akun.

Kebijakan IAM dan SCP

Anda dapat menggunakan kebijakan IAM atau kebijakan kontrol AWS Organizations layanan (SCP) untuk mengontrol pengguna sebagai berikut:

- Tidak dapat meluncurkan instance menggunakan RunInstances API kecuali instans dikonfigurasi untuk menggunakan IMDSv2.
- Tidak dapat mengubah instance yang ada menggunakan ModifyInstanceMetadataOptions API untuk mengaktifkan kembali IMDSv1.

Kebijakan IAM atau SCP harus berisi kunci syarat IAM berikut:

- `ec2:MetadataHttpEndpoint`
- `ec2:MetadataHttpPutResponseHopLimit`
- `ec2:MetadataHttpTokens`

Jika parameter dalam panggilan API atau CLI tidak cocok dengan status yang ditentukan dalam kebijakan yang berisi kunci kondisi, panggilan API atau CLI gagal dengan respons`UnauthorizedOperation`.

Selain itu, Anda dapat memilih lapisan perlindungan tambahan untuk menegakkan perubahan dari IMDSv1 ke IMDSv2. Pada lapisan manajemen akses sehubungan dengan API yang dipanggil melalui kredentif Peran EC2, Anda dapat menggunakan kunci kondisi baik dalam kebijakan IAM atau kebijakan kontrol AWS Organizations layanan (SCP). Secara khusus, dengan menggunakan

kunci syarat `ec2:RoleDelivery` dengan nilai `2.0` dalam kebijakan IAM Anda, panggilan API yang dilakukan dengan kredenal Peran EC2 yang diperoleh dari IMDSv1 akan menerima tanggapan `UnauthorizedOperation`. Hal yang sama dapat dicapai secara lebih luas dengan kondisi yang disyaratkan oleh SCP. Ini memastikan bahwa kredenSIAL yang dikirimkan melalui IMDSv1 sebenarnya tidak dapat digunakan untuk memanggil API karena panggilan API apa pun yang tidak cocok dengan kondisi yang ditentukan akan menerima kesalahan. `UnauthorizedOperation`

Untuk contoh kebijakan IAM, lihat Cara menggunakan metadata instans. Untuk informasi selengkapnya tentang SCP, lihat Kebijakan kontrol layanan di Panduan AWS Organizations Pengguna.

Kebijakan Deklaratif

Gunakan Kebijakan Deklaratif (fitur AWS Organizations) untuk mengatur default akun IMDS secara terpusat, termasuk penegakan IMDSv2, di seluruh organisasi Anda. Untuk contoh kebijakan, lihat tab Metadata Instance di bagian Kebijakan deklaratif yang didukung di Panduan AWS Organizations Pengguna.

Jalur yang disarankan untuk mengharuskan IMDSv2

Menggunakan alat di atas, kami merekomendasikan jalur berikut untuk transisi ke IMDSv2:

- Langkah 1: Identifikasi instance dengan IMDSv2 = opsional dan audit penggunaan IMDSv1
- Langkah 2: Perbarui perangkat lunak ke IMDSv2
- Langkah 3: Memerlukan IMDSv2 pada instance
- Langkah 4: Setel IMDSv2 = Diperlukan sebagai default
- Langkah 5: Menerapkan instans untuk memerlukan IMDSv2

Langkah 1: Identifikasi instance dengan IMDSv2 = opsional dan audit penggunaan IMDSv1

Untuk menilai cakupan migrasi IMDSv2 Anda, identifikasi instance yang dikonfigurasi untuk mengizinkan IMDSv1 atau IMDSv2, dan audit panggilan IMDSv1.

1. Identifikasi instance yang dikonfigurasi untuk mengizinkan IMDSv1 atau IMDSv2:

   Amazon EC2 console

   1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Untuk melihat hanya instance yang dikonfigurasi untuk mengizinkan IMDSv1 atau IMDSv2, tambahkan filter IMDSv2 = opsional.
4. Atau, untuk melihat apakah IMDSv2 op sional atau diperlukan untuk semua instance, buka jendela Preferensi (ikon roda gigi), aktifkan IMDSv2, dan pilih Konfirmasi. Ini menambahkan kolom IMDSv2 ke tabel Instances.

AWS CLI

Gunakan perintah describe-instance dan filter berdasarkan`metadata-options.http-tokens = optional`, sebagai berikut:

```
aws ec2 describe-instances --filters "Name=metadata-options.http-
tokens,Values=optional" --query "Reservations[*].Instances[*].[InstanceId]" --
output text
```

2. Audit panggilan IMDSv1 pada setiap instance:

   Gunakan met CloudWatch rik`MetadataNoToken`. Metrik ini melacak jumlah panggilan IMDSv1 ke IMDS di instans Anda. Untuk informasi selengkapnya, lihat Metrik instans.

3. Identifikasi perangkat lunak pada instans Anda yang membuat panggilan IMDSv1:

   Gunakan IMDS Packet Analyzer sumber terbuka untuk mengidentifikasi dan mencatat panggilan IMDSv1 selama fase boot instans dan operasi runtime Anda. Gunakan informasi ini untuk mengidentifikasi perangkat lunak yang akan diperbarui agar instans Anda siap menggunakan IMDSv2 saja. Anda dapat menjalankan IMDS Packet Analyzer dari baris perintah atau menginstalnya sebagai layanan.

Langkah 2: Perbarui perangkat lunak ke IMDSv2

Perbarui semua SDK, CLI, dan perangkat lunak yang menggunakan kredentif Peran pada instans Anda ke IMDSv2-compatible versi. Untuk informasi selengkapnya tentang memperbarui CLI, lihat Meng instal atau memperbarui ke versi terbaru AWS CLI di Panduan AWS Command Line Interface Pengguna.

Langkah 3: Memerlukan IMDSv2 pada instance

Setelah mengonfirmasi nol panggilan IMDSv1 melalui `MetadataNoToken` metrik, konfigurasikan instans yang ada untuk memerlukan IMDSv2. Juga, konfigurasikan semua instance baru untuk memerlukan IMDSv2. Dengan kata lain, nonaktifkan IMDSv1 pada semua instance yang ada dan baru.

1. Konfigurasikan instans yang ada untuk memerlukan IMDSv2:

   Amazon EC2 console

   1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
   2. Di panel navigasi, pilih Instans.
   3. Pilih instans Anda.
   4. Pilih Tindakan, Pengaturan instans, Ubah opsi metadata instans.
   5. Untuk IMDSv2, pilih Wajib
   6. Pilih Simpan.

   AWS CLI

   Gunakan perintah CLI modify-instance-metadata-options untuk menentukan bahwa hanya IMDSv2 yang akan digunakan.

   > **Note**
   >
   > Anda dapat mengubah pengaturan ini pada instans yang sedang berjalan. Perubahan mulai berlaku segera tanpa perlu restart instance.

   Untuk informasi selengkapnya, lihat Mengharuskan penggunaan IMDSv2.
2. Pantau masalah setelah menonaktifkan IMDSv1:

   1. Lacak berapa kali panggilan IMDSv1 dicoba dan ditolak dengan metrik. `MetadataNoTokenRejected` CloudWatch
   2. Jika `MetadataNoTokenRejected` metrik mencatat panggilan IMDSv1 pada instance yang mengalami masalah perangkat lunak, ini menunjukkan bahwa perangkat lunak memerlukan pembaruan untuk menggunakan IMDSv2.

3. Konfigurasikan instans baru untuk memerlukan IMDSv2:

   Amazon EC2 console

   1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
   2. Ikuti langkah-langkah untuk meluncurkan instance.
   3. Perluas Detail lanj utan, dan untuk versi Metadata, pilih V2 saja (token diperlukan).
   4. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans.

      Untuk informasi selengkapnya, lihat Konfigurasikan instans saat peluncuran.

   AWS CLI

   AWS CLI: Gunakan perintah run-instance dan tentukan bahwa IMDSv2 diperlukan.

Langkah 4: Setel IMDSv2 = Diperlukan sebagai default

Anda dapat mengatur IMDSV2=wajib sebagai konfigurasi default baik di tingkat akun atau organisasi. Ini memastikan bahwa semua instans yang baru diluncurkan secara otomatis dikonfigurasi untuk memerlukan IMDSv2.

1. Tetapkan default tingkat akun:

   Amazon EC2 console

   1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
   2. Di panel navigasi, pilih Dasbor.
   3. Pada kartu Atribut Akun, di bawah Pengaturan, pilih Perlindungan dan keamanan data.
   4. Di bawah IMDS default, pilih Kelola.
   5. Untuk layanan metadata contoh, pilih Di aktifkan.
   6. Untuk versi Metadata, pilih V2 saja (diperlukan token).
   7. Pilih Perbarui.

   AWS CLI

   Gunakan perintah CLI modify-instance-metadat a-dedefault dan tentukan dan. `--http-tokens required --http-put-response-hop-limit 2`

Untuk informasi selengkapnya, lihat Setel IMDSv2 sebagai default untuk akun.

2. Atau, atur default tingkat organisasi menggunakan Kebijakan Deklaratif:

   Gunakan Kebijakan Deklaratif untuk mengatur default organisasi untuk IMDSv2 menjadi wajib. Untuk contoh kebijakan, lihat tab Metadata Instance di bagian Kebijakan deklaratif yang didukung di Panduan AWS Organizations Pengguna.

Langkah 5: Menerapkan instans untuk memerlukan IMDSv2

Setelah Anda mengonfirmasi bahwa tidak ada ketergantungan pada IMDSv1 pada instans mana pun, sebaiknya Anda menerapkan IMDSv2 pada semua instans baru.

Gunakan salah satu opsi berikut untuk menerapkan IMDSv2:

1. Menegakkan IMDSv2 dengan properti akun

   Anda dapat menerapkan penggunaan IMDSv2 di tingkat akun untuk masing-masing. Wilayah AWS Saat diterapkan, instance hanya dapat diluncurkan jika dikonfigurasi untuk memerlukan IMDSv2. Penegakan ini berlaku terlepas dari bagaimana instance atau AMI dikonfigurasi. Untuk informasi selengkapnya, lihat Menerapkan IMDSv2 di tingkat akun. Untuk menerapkan pengaturan ini di tingkat organisasi, tetapkan Kebijakan Deklaratif. Untuk contoh kebijakan, lihat tab Metadata Instance di bagian Kebijakan deklaratif yang didukung di Panduan AWS Organizations Pengguna.

   Untuk mencegah pembalikan penegakan, Anda harus menggunakan kebijakan IAM untuk mencegah akses ke ModifyInstanceMetadataDefaults API. Untuk informasi selengkapnya, lihat Gunakan kebijakan IAM.

   > **Note**
   >
   > Pengaturan ini tidak mengubah versi IMDS dari instans yang ada, tetapi memblokir mengaktifkan IMDSv1 pada instans yang saat ini menonaktifkan IMDSv1.

> **Warning**
>
> Jika penegakan IMDSv2 diaktifkan dan `httpTokens` tidak disetel ke `required` konfigurasi instans saat peluncuran, pengaturan akun, atau konfigurasi AMI, peluncuran instance akan gagal. Untuk informasi pemecahan masalah, lihat Meluncurkan IMDSv1-enabled instance gagal.

2. Atau, terapkan IMDSv2 dengan menggunakan kunci kondisi IAM atau SCP berikut:
   - `ec2:MetadataHttpTokens`
   - `ec2:MetadataHttpPutResponseHopLimit`
   - `ec2:MetadataHttpEndpoint`

   Kunci kondisi ini mengontrol penggunaan RunInstances dan ModifyInstanceMetadataOptions API serta CLI yang sesuai. Jika kebijakan dibuat, dan parameter dalam panggilan API tidak cocok dengan status yang ditentukan dalam kebijakan menggunakan kunci syarat, panggilan API atau CLI akan gagal dengan tanggapan`UnauthorizedOperation`.

   Misalnya kebijakan IAM, lihat .Cara menggunakan metadata instans

## Batasi akses ke Layanan Metadata Instance

Anda dapat mempertimbangkan untuk menggunakan aturan firewall lokal untuk menonaktifkan akses dari beberapa atau semua proses ke Instance Metadata Service (IMDS).

Misalnya, IMDS dapat dijangkau dari jaringan Anda sendiri ketika alat jaringan dalam VPC Anda, seperti router virtual, meneruskan paket ke alamat IMDS, dan source/destination pemeriksaan default pada Nitro-based instance dinonaktifkan. Untuk mencegah sumber dari luar VPC Anda mencapai IMDS, sebaiknya Anda memodifikasi konfigurasi alat jaringan untuk menghapus paket dengan alamat IPv4 tujuan IMDS `169.254.169.254` dan, jika Anda mengaktifkan titik akhir IPv6, alamat IPv6 dari IMDS. `[fd00:ec2::254]`

Batasi akses IMDS untuk instans Linux

Menggunakan iptables untuk membatasi akses

Contoh berikut menggunakan iptables Linux dan modul `owner` untuk mencegah server web Apache (berdasarkan ID pengguna instalasi default `apache`) mengakses 169.254.169.254. Contoh ini

menggunakan aturan penolakan untuk menolak semua permintaan metadata instans (baik IMDSv1 atau IMDSv2) dari proses apa pun yang berjalan sebagai pengguna itu.

```
$ sudo iptables --append OUTPUT --proto tcp --destination 169.254.169.254 --match owner
 --uid-owner apache --jump REJECT
```

Atau, Anda dapat mempertimbangkan untuk hanya mengizinkan akses ke pengguna atau grup tertentu, dengan menggunakan aturan izin. Aturan izinkan mungkin lebih mudah dikelola dari perspektif keamanan, karena aturan tersebut mengharuskan Anda membuat keputusan tentang perangkat lunak apa yang memerlukan akses ke metadata instans. Jika Anda menggunakan aturan izin, kecil kemungkinannya Anda secara tidak sengaja mengizinkan perangkat lunak mengakses layanan metadata (yang tidak Anda inginkan untuk mempunyai akses) jika nanti Anda mengubah perangkat lunak atau konfigurasi pada sebuah instans. Anda juga dapat menggabungkan penggunaan grup dengan aturan izin, sehingga Anda dapat menambahkan dan menghapus pengguna dari grup yang diizinkan tanpa perlu mengubah aturan firewall.

Contoh berikut mencegah akses ke IMDS oleh semua proses, kecuali untuk proses yang berjalan di akun pengguna `trustworthy-user`.

```
$ sudo iptables --append OUTPUT --proto tcp --destination 169.254.169.254 --match
 owner ! --uid-owner trustworthy-user --jump REJECT
```

> **Note**
>
> - Untuk menggunakan aturan firewall lokal, Anda perlu menyesuaikan perintah contoh sebelumnya agar sesuai dengan kebutuhan Anda.
> - Secara default, aturan iptables tidak persisten di seluruh boot ulang sistem. Aturan itu dapat dibuat persisten dengan menggunakan fitur OS, yang tidak dijelaskan di sini.
> - Modul `owner` iptables hanya cocok dengan keanggotaan grup jika grup tersebut adalah grup utama dari pengguna lokal tertentu. Grup lain tidak cocok.

### Menggunakan PF atau IPFW untuk membatasi akses

Jika Anda menggunakanFreeBSD atauOpenBSD , Anda juga dapat mempertimbangkan untuk menggunakan PF atau IPFW. Contoh berikut membatasi akses ke IMDS hanya untuk pengguna root.

PF

```
$ block out inet proto tcp from any to 169.254.169.254
```

```
$ pass out inet proto tcp from any to 169.254.169.254 user root
```

IPFW

```
$ allow tcp from any to 169.254.169.254 uid root
```

```
$ deny tcp from any to 169.254.169.254
```

> **Note**
>
> Urutan perintah PF dan IPFW penting. PF default ke aturan pencocokan terakhir dan IPFW default ke aturan pencocokan pertama.

Batasi akses IMDS untuk instans Windows

Menggunakan firewall Windows untuk membatasi akses

PowerShell Contoh berikut menggunakan firewall Windows bawaan untuk mencegah server web Server Informasi Internet (berdasarkan ID pengguna instalasi defaultNT AUTHORITY\IUSR) mengakses 169.254.169.254. Contoh ini menggunakan aturan penolakan untuk menolak semua permintaan metadata instans (baik IMDSv1 atau IMDSv2) dari proses apa pun yang berjalan sebagai pengguna itu.

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount ("NT
 AUTHORITY\IUSR")
PS C:\> $BlockPrincipalSID =
 $blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $BlockPrincipalSDDL = "D:(A;;CC;;;$BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service from IIS" -Action
 block -Direction out `
-Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $BlockPrincipalSDDL
```

Atau, Anda dapat mempertimbangkan untuk hanya mengizinkan akses ke pengguna atau grup tertentu, dengan menggunakan aturan izin. Aturan izinkan mungkin lebih mudah dikelola dari perspektif keamanan, karena aturan tersebut mengharuskan Anda membuat keputusan tentang

perangkat lunak apa yang memerlukan akses ke metadata instans. Jika Anda menggunakan aturan izin, kecil kemungkinannya Anda secara tidak sengaja mengizinkan perangkat lunak mengakses layanan metadata (yang tidak Anda inginkan untuk mempunyai akses) jika nanti Anda mengubah perangkat lunak atau konfigurasi pada sebuah instans. Anda juga dapat menggabungkan penggunaan grup dengan aturan izin, sehingga Anda dapat menambahkan dan menghapus pengguna dari grup yang diizinkan tanpa perlu mengubah aturan firewall.

Contoh berikut mencegah akses ke metadata instans oleh semua proses yang berjalan sebagai grup OS yang ditentukan dalam variabel `blockPrincipal` (dalam contoh ini, grup Windows `Everyone`), kecuali untuk proses yang ditentukan dalam `exceptionPrincipal` (dalam contoh ini, grup yang bernama `trustworthy-users`). Anda harus menentukan baik menolak maupun mengizinkan pengguna utama karena Windows Firewall, tidak seperti aturan `! --uid-owner trustworthy-user` di iptables Linux, tidak menyediakan mekanisme pintasan untuk mengizinkan hanya pengguna utama (pengguna atau grup) tertentu dengan menolak semua yang lain.

```
PS C:\> $blockPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
 ("Everyone")
PS C:\> $BlockPrincipalSID =
 $blockPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $exceptionPrincipal = New-Object -TypeName System.Security.Principal.NTAccount
 ("trustworthy-users")
PS C:\> $ExceptionPrincipalSID =
 $exceptionPrincipal.Translate([System.Security.Principal.SecurityIdentifier]).Value
PS C:\> $PrincipalSDDL = "O:LSD:(D;;CC;;;$ExceptionPrincipalSID)(A;;CC;;;
$BlockPrincipalSID)"
PS C:\> New-NetFirewallRule -DisplayName "Block metadata service for
 $($blockPrincipal.Value), exception: $($exceptionPrincipal.Value)" -Action block -
Direction out `
-Protocol TCP -RemoteAddress 169.254.169.254 -LocalUser $PrincipalSDDL
```

> **Note**
>
> Untuk menggunakan aturan firewall lokal, Anda perlu menyesuaikan perintah contoh sebelumnya agar sesuai dengan kebutuhan Anda.

### Menggunakan aturan netsh untuk membatasi akses

Anda dapat mempertimbangkan untuk memblokir semua perangkat lunak menggunakan aturan `netsh`, tetapi itu sangat kurang fleksibel.

```
C:\> netsh advfirewall firewall add rule name="Block metadata service altogether"
 dir=out protocol=TCP remoteip=169.254.169.254 action=block
```

Note

- Untuk menggunakan aturan firewall lokal, Anda perlu menyesuaikan perintah contoh sebelumnya agar sesuai dengan kebutuhan Anda.
- Aturan `netsh` harus disetel dari command prompt yang tinggi, dan tidak dapat diatur untuk menolak atau mengizinkan pengguna utama tertentu.

## Mengkonfigurasi opsi Layanan Metadata Instance

Instance Metadata Service (IMDS) berjalan secara lokal pada setiap instans EC2. Opsi metadata instans  mengacu pada sekumpulan konfigurasi yang mengontrol aksesibilitas dan perilaku IMDS pada instans EC2.

Anda dapat mengonfigurasi opsi metadata instance berikut pada setiap instance:

Layanan metadata instans (IMDS): | `enabled` `disabled`

Anda dapat mengaktifkan atau menonaktifkan IMDS pada instance. Saat dinonaktifkan, Anda atau kode apa pun tidak akan dapat mengakses metadata instans pada instance.

IMDS memiliki dua titik akhir pada sebuah instance: IPv4 (`169.254.169.254`) dan IPv6 (). `[fd00:ec2::254]` Saat Anda mengaktifkan IMDS, titik akhir IPv4 secara otomatis diaktifkan. Jika Anda ingin mengaktifkan titik akhir IPv6, Anda harus melakukannya secara eksplisit.

Titik akhir IPv6 IMDS: | `enabled` `disabled`

Anda dapat secara eksplisit mengaktifkan titik akhir IPv6 IMDS pada sebuah instance. Ketika titik akhir IPv6 diaktifkan, titik akhir IPv4 tetap diaktifkan. Titik akhir IPv6 hanya didukung pada Nitro-based instance di IPv6-supported subnet (dual stack atau IPv6 saja).

Versi metadata: `IMDSv1 or IMDSv2 (token optional)` | `IMDSv2 only (token required)`

Saat meminta metadata instans, panggilan IMDSv2 memerlukan token. Panggilan IMDSv1 tidak memerlukan token. Anda dapat mengonfigurasi instance untuk mengizinkan panggilan IMDSv1 atau IMDSv2 (di mana token bersifat opsional), atau untuk hanya mengizinkan panggilan IMDSv2 (jika token diperlukan).

Batas hop respons metadata: 1 — 64

Batas hop adalah jumlah lompatan jaringan yang diizinkan untuk dilakukan oleh respons PUT. Anda dapat mengatur batas hop ke minimum 1 dan maksimum64. Dalam lingkungan wadah, batas hop 1 dapat menyebabkan masalah. Untuk informasi tentang cara mengurangi masalah ini, lihat informasi tentang lingkungan kontainer di bawahPertimbangan akses metadata instans.

Akses ke tag dalam metadata contoh: `enabled` | `disabled`

Anda dapat mengaktifkan atau menonaktifkan akses ke tag instans dari metadata instans. Untuk informasi selengkapnya, lihat Melihat tag untuk instans EC2 menggunakan metadata instans.

Untuk melihat konfigurasi instans saat ini, lihatMengueri opsi metadata instans untuk instans yang ada.

## Tempat mengonfigurasi opsi metadata instance

Opsi metadata instans dapat dikonfigurasi pada tingkat yang berbeda, sebagai berikut:

- Akun — Anda dapat menetapkan nilai default untuk opsi metadata instans di tingkat akun untuk masing-masing Wilayah AWS. Saat instance diluncurkan, opsi metadata instance secara otomatis disetel ke nilai tingkat akun. Anda dapat mengubah nilai-nilai ini saat peluncuran. Account-level nilai default tidak mempengaruhi instance yang ada.
- AMI — Anda dapat mengatur `imds-support` parameter ke `v2.0` saat Anda mendaftar atau memodifikasi AMI. Ketika sebuah instance diluncurkan dengan AMI ini, versi metadata instance secara otomatis disetel ke IMDSv2 dan batas hop diatur ke 2.
- Inst ance — Anda dapat mengubah semua opsi metadata instance pada instance saat diluncurkan, mengesampingkan pengaturan default. Anda juga dapat mengubah opsi metadata instance setelah diluncurkan pada instance yang sedang berjalan atau dihentikan. Perhatikan bahwa perubahan mungkin dibatasi oleh kebijakan IAM atau SCP.

Untuk informasi selengkapnya, lihat Mengonfigurasi opsi metadata instans untuk instans baru dan Mengonfigurasi opsi metadata instans untuk instans yang ada.

## Urutan prioritas misalnya opsi metadata

Nilai untuk setiap opsi metadata instance ditentukan saat peluncuran instance, mengikuti urutan prioritas hierarkis. Hirarki, dengan prioritas tertinggi di atas, adalah sebagai berikut:

- Prioritas 1: Konfigurasi instans saat peluncuran — Nilai dapat ditentukan baik di template peluncuran atau dalam konfigurasi instance. Setiap nilai yang ditentukan di sini menggantikan nilai yang ditentukan pada tingkat akun atau di AMI.
- Prioritas 2: Pengaturan akun — Jika nilai tidak ditentukan saat peluncuran instance, maka itu ditentukan oleh pengaturan tingkat akun (yang ditetapkan untuk masing-masing). Wilayah AWS Account-levelpengaturan termasuk nilai untuk setiap opsi metadata, atau menunjukkan tidak ada preferensi sama sekali.
- Prioritas 3: Konfigur asi AMI — Jika nilai tidak ditentukan saat peluncuran instance atau di tingkat akun, maka itu ditentukan oleh konfigurasi AMI. Ini hanya berlaku untuk `HttpTokens` dan`HttpPutResponseHopLimit`.

Setiap opsi metadata dievaluasi secara terpisah. Instans dapat dikonfigurasi dengan campuran konfigurasi instans langsung, default tingkat akun, dan konfigurasi dari AMI.

Anda dapat mengubah nilai opsi metadata apa pun setelah diluncurkan pada instans yang sedang berjalan atau dihentikan, kecuali perubahan dibatasi oleh kebijakan IAM atau SCP.

> **Note**
>
> Setelan penegakan IMDSv2 tingkat akun dievaluasi setelah urutan prioritas menentukan pengaturan IMDS instans. Ketika penegakan IMDSv2 diaktifkan, instans yang diaktifkan dengan IMDSv1 akan gagal. Untuk informasi lebih lanjut tentang penegakan hukum, lihatMenerapkan IMDSv2 di tingkat akun.

> **Warning**
>
> Jika penegakan IMDSv2 diaktifkan dan `httpTokens` belum disetel ke `required` konfigurasi instans saat peluncuran, pengaturan akun, atau konfigurasi AMI, peluncuran Anda akan gagal.

Contoh 1 — Tentukan nilai untuk opsi metadata

Dalam contoh ini, instance EC2 diluncurkan ke Wilayah yang `HttpPutResponseHopLimit` disetel ke 1 tingkat akun. AMI yang `ImdsSupport` ditentukan telah disetel ke`v2.0`. Tidak ada opsi

metadata yang ditentukan langsung pada instance saat peluncuran. Instance diluncurkan dengan opsi metadata berikut:

```
"MetadataOptions": {
    ...
    "HttpTokens": "required",
    "HttpPutResponseHopLimit": 1,
    ...
```

Nilai-nilai ini ditentukan sebagai berikut:

- Tidak ada opsi metadata yang ditentukan saat peluncuran: Selama peluncuran instance, nilai spesifik untuk opsi metadata tidak disediakan baik dalam parameter peluncuran instance atau dalam template peluncuran.
- Pengaturan akun diutamakan berikutnya: Jika tidak ada nilai tertentu yang ditentukan saat peluncuran, pengaturan di tingkat akun dalam Wilayah akan diutamakan. Ini berarti bahwa nilai default yang dikonfigurasi pada tingkat akun diterapkan. Dalam hal ini, `HttpPutResponseHopLimit` ditetapkan ke`1`.
- Pengaturan AMI diutamakan terakhir: Jika tidak ada nilai tertentu yang ditentukan saat peluncuran atau pada tingkat akun untuk `HttpTokens` (versi metadata instance), pengaturan AMI diterapkan. Dalam hal ini, pengaturan AMI `ImdsSupport: v2.0` menentukan yang `HttpTokens` disetel ke`required`. Perhatikan bahwa meskipun pengaturan AMI `ImdsSupport: v2.0` dirancang untuk disetel`HttpPutResponseHopLimit: 2`, pengaturan ini diganti oleh pengaturan tingkat akun`HttpPutResponseHopLimit: 1`, yang memiliki prioritas lebih tinggi.

Contoh 2 — Tentukan nilai untuk opsi metadata

Dalam contoh ini, instance EC2 diluncurkan dengan pengaturan yang sama seperti pada Contoh 1 sebelumnya, tetapi dengan disetel ke `HttpTokens optional` langsung pada instance saat peluncuran. Instance diluncurkan dengan opsi metadata berikut:

```
"MetadataOptions": {
    ...
    "HttpTokens": "optional",
    "HttpPutResponseHopLimit": 1,
    ...
```

Nilai untuk `HttpPutResponseHopLimit` ditentukan dengan cara yang sama seperti pada Contoh 1. Namun, nilai untuk `HttpTokens` ditentukan sebagai berikut: Opsi metadata yang dikonfigurasi pada instance saat peluncuran diutamakan terlebih dahulu. Meskipun AMI dikonfigurasi dengan `ImdsSupport: v2.0` (dengan kata lain, disetel ke`required`), `HttpTokens` nilai yang ditentukan pada instance saat peluncuran (`HttpTokens`disetel ke`optional`) diutamakan.

Contoh 3 — Tentukan nilai untuk opsi metadata dengan HttpTokensEnforced diaktifkan

Dalam contoh ini, akun di Wilayah memiliki `HttpTokens = required` dan`HttpTokensEnforced = enabled`.

Pertimbangkan upaya peluncuran instans EC2 berikut:

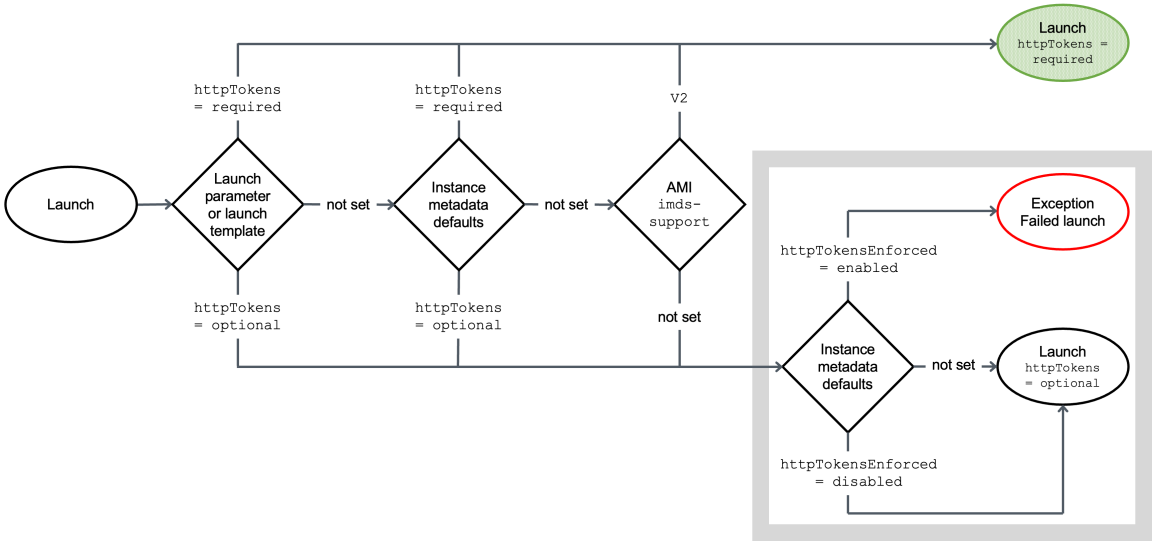
- Upaya peluncuran dengan `HttpTokens` set ke `optional` — Peluncuran gagal karena penegakan tingkat akun diaktifkan (`HttpTokensEnforced = enabled`) dan parameter peluncuran diutamakan daripada default akun.
- Upaya peluncuran dengan `HttpTokens` set ke `required` — Peluncuran berhasil karena sesuai dengan penegakan tingkat akun.
- Upaya peluncuran tanpa `HttpTokens` nilai yang ditentukan — Peluncuran berhasil karena nilai default `required` berdasarkan pengaturan akun.

Mengatur versi metadata instance

Saat sebuah instance diluncurkan, nilai untuk versi metadata instans adalah IMDSv1 atau IMDSv2 (token opsional) () atau hanya IMDSv2 (token requiredhttpTokens=optional) (). **httpTokens=required**

Saat peluncuran instance, Anda dapat secara manual menentukan nilai untuk versi metadata, atau menggunakan nilai default. Jika Anda menentukan nilai secara manual, nilainya akan menggantikan default apa pun. Jika Anda memilih untuk tidak menentukan nilai secara manual, itu akan ditentukan oleh kombinasi pengaturan default.

Diagram alir berikut menunjukkan bagaimana versi metadata untuk instance saat peluncuran ditentukan oleh pengaturan pada tingkat konfigurasi yang berbeda dan di mana penegakan dievaluasi. Tabel berikut menyediakan pengaturan spesifik di setiap level.




Tabel menunjukkan bagaimana versi metadata untuk instance saat peluncuran (ditunjukkan oleh Konfigurasi instance yang dihasilkan di kolom 4) ditentukan oleh pengaturan pada tingkat konfigurasi yang berbeda. Urutan prioritas adalah dari kiri ke kanan, di mana kolom pertama mengambil prioritas tertinggi, sebagai berikut:

- Kolom 1: Parameter peluncuran — Merupakan pengaturan pada instance yang Anda tentukan secara manual saat peluncuran.
- Kolom 2: Standar tingkat akun — Merupakan pengaturan untuk akun.
- Kolom 3: AMI default - Merupakan pengaturan pada AMI.

| Parameter luncurkan | Tingkat akun default | AMI default | Konfigurasi instance yang dihasilkan |
|---|---|---|---|
| Hanya V2 (diperlukan token) | Tidak ada preferensi | Hanya V2 | Hanya V2 |
| Hanya V2 (diperlukan token) | Hanya V2 | Hanya V2 | Hanya V2 |
| Hanya V2 (diperlukan token) | V1 atau V2 | Hanya V2 | Hanya V2 |
| V1 atau V2 (token opsional) | Tidak ada preferensi | Hanya V2 | V1 atau V2 |

| Parameter luncurkan | Tingkat akun default | AMI default | Konfigurasi instance yang dihasilkan |
|---|---|---|---|
| V1 atau V2 (token opsional) | Hanya V2 | Hanya V2 | V1 atau V2 |
| V1 atau V2 (token opsional) | V1 atau V2 | Hanya V2 | V1 atau V2 |
| Tidak disetel | Tidak ada preferensi | Hanya V2 | Hanya V2 |
| Tidak disetel | Hanya V2 | Hanya V2 | Hanya V2 |
| Tidak disetel | V1 atau V2 | Hanya V2 | V1 atau V2 |
| Hanya V2 (diperlukan token) | Tidak ada preferensi | null | Hanya V2 |
| Hanya V2 (diperlukan token) | Hanya V2 | null | Hanya V2 |
| Hanya V2 (diperlukan token) | V1 atau V2 | null | Hanya V2 |
| V1 atau V2 (token opsional) | Tidak ada preferensi | null | V1 atau V2 |
| V1 atau V2 (token opsional) | Hanya V2 | null | V1 atau V2 |
| V1 atau V2 (token opsional) | V1 atau V2 | null | V1 atau V2 |
| Tidak disetel | Tidak ada preferensi | null | V1 atau V2 |
| Tidak disetel | Hanya V2 | null | Hanya V2 |
| Tidak disetel | V1 atau V2 | null | V1 atau V2 |

## Gunakan kunci kondisi IAM untuk membatasi opsi metadata instance

Anda dapat menggunakan kunci kondisi IAM dalam kebijakan IAM atau SCP sebagai berikut:

- Izinkan sebuah instans untuk diluncurkan hanya jika instans tersebut dikonfigurasi untuk mengharuskan penggunaan IMDSv2
- Batasi jumlah hop yang diizinkan
- Nonaktifkan akses untuk metadata instans

Tugas

- Mengonfigurasi opsi metadata instans untuk instans baru
- Mengonfigurasi opsi metadata instans untuk instans yang ada

> **Note**
>
> Anda harus melanjutkan dengan hati-hati dan melakukan pengujian yang cermat sebelum membuat perubahan apa pun. Perhatikan hal-hal berikut ini:
>
> - Jika Anda memaksakan penggunaan IMDSv2, aplikasi atau agen yang menggunakan IMDSv1 untuk akses metadata instans akan rusak.
> - Jika Anda menonaktifkan semua akses ke metadata instans, aplikasi atau agen yang mengandalkan akses metadata instans ke fungsi akan rusak.
> - Untuk IMDSv2, Anda harus menggunakan `/latest/api/token` saat mengambil token.
> - (Hanya Windows) Jika PowerShell versi Anda lebih awal dari 4.0, Anda harus memperbarui ke Windows Management Framework 4.0 untuk memerlukan penggunaan IMDSv2.

## Mengonfigurasi opsi metadata instans untuk instans baru

Anda dapat mengonfigurasi opsi metadata instance berikut untuk instans baru.

Opsi

- Mengharuskan penggunaan IMDSv2
- Aktifkan titik akhir IPv4 dan IPv6 IMDS

- Nonaktifkan akses untuk metadata instans
- Mengizinkan akses ke tanda dalam metadata instans

> **Note**
>
> Pengaturan untuk opsi ini dikonfigurasi di tingkat akun, baik langsung di akun atau dengan menggunakan kebijakan deklaratif. Mereka harus dikonfigurasi di setiap Wilayah AWS tempat Anda ingin mengonfigurasi opsi metadata instance. Menggunakan kebijakan deklaratif memungkinkan Anda menerapkan pengaturan di beberapa Wilayah secara bersamaan, serta di beberapa akun secara bersamaan. Saat kebijakan deklaratif digunakan, Anda tidak dapat mengubah pengaturan secara langsung di dalam akun. Topik ini menjelaskan cara mengonfigurasi pengaturan secara langsung di dalam akun. Untuk informasi tentang penggunaan kebijakan deklaratif, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Mengharuskan penggunaan IMDSv2

Anda dapat menggunakan metode berikut untuk mewajibkan penggunaan IMDSv2 pada instans baru Anda.

Untuk mengharuskan IMDSv2

- Setel IMDSv2 sebagai default untuk akun
- Menerapkan IMDSv2 di tingkat akun
- Konfigurasikan instans saat peluncuran
- Konfigurasikan AMI
- Gunakan kebijakan IAM

Setel IMDSv2 sebagai default untuk akun

Anda dapat mengatur versi default untuk layanan metadata instans (IMDS) di tingkat akun untuk masing-masing Wilayah AWS. Ini berarti bahwa ketika Anda meluncurkan instance baru, versi metadata instans secara otomatis disetel ke default tingkat akun. Namun, Anda dapat mengganti nilai secara manual saat peluncuran atau setelah peluncuran. Untuk informasi selengkapnya tentang bagaimana pengaturan tingkat akun dan penggantian manual memengaruhi instance, lihat. Urutan prioritas misalnya opsi metadata

> **Note**
>
> Menyetel default tingkat akun tidak mengatur ulang instans yang ada. Misalnya, jika Anda menyetel default tingkat akun ke IMDSv2, instans yang ada yang disetel ke IMDsv1 tidak terpengaruh. Jika Anda ingin mengubah nilai pada instance yang ada, Anda harus mengubah nilai pada instans itu sendiri secara manual.

Anda dapat mengatur default akun untuk versi metadata instance ke IMDSv2 sehingga semua instans baru dalam peluncuran akun dengan IMDSv2 diperlukan, dan IMDSv1 akan dinonaktifkan. Dengan default akun ini, saat Anda meluncurkan instance, berikut ini adalah nilai default untuk instance:

- Konsol: Versi metadata diatur ke V2 saja (diperlukan token) dan batas hop respons Metadata diatur ke 2.
- AWS CLI: `HttpTokens` disetel ke `required` dan `HttpPutResponseHopLimit` disetel ke2.

> **Note**
>
> Sebelum menyetel default akun ke IMDSv2, pastikan instans Anda tidak bergantung pada IMDSv1. Untuk informasi selengkapnya, lihat Jalur yang disarankan untuk mengharuskan IMDSv2.

Console

Untuk mengatur IMDSv2 sebagai default untuk akun untuk Wilayah yang ditentukan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Untuk mengubah Wilayah AWS, gunakan pemilih Wilayah di sudut kanan atas halaman.
3. Di panel navigasi, pilih Dasbor.
4. Pada kartu Atribut Akun, di bawah Pengaturan, pilih Perlindungan dan keamanan data.
5. Di samping IMDS default, pilih Kelola.
6. Pada halaman Kelola default IMDS, lakukan hal berikut:

   a. Untuk layanan metadata contoh, pilih Di aktifkan.

b. Untuk Versi metadata, pilih V2 saja (token diperlukan).

c. Untuk batas hop respons Metadata, tentukan 2 jika instans Anda akan meng-host wadah. Jika tidak, pilih Tidak ada preferensi. Ketika tidak ada preferensi yang ditentukan, saat peluncuran, nilai default menjadi 2 jika AMI memiliki pengaturan`ImdsSupport: v2.0`; jika tidak, default ke 1.

d. Pilih Perbarui.

AWS CLI

Untuk mengatur IMDSv2 sebagai default untuk akun untuk Wilayah yang ditentukan

Gunakan perintah modify-instance-metadata-dedefault dan tentukan Wilayah tempat untuk memodifikasi pengaturan tingkat akun IMDS. Sertakan `--http-tokens` disetel ke `required` dan `--http-put-response-hop-limit` setel ke 2 jika instans Anda akan meng-host kontainer. Jika tidak, tent `-1` ukan untuk menunjukkan tidak ada preferensi. Ketika `-1` (tidak ada preferensi) ditentukan, saat peluncuran, nilai default adalah 2 jika AMI memiliki pengaturan`ImdsSupport: v2.0`; jika tidak maka defaultnya menjadi. 1

```
aws ec2 modify-instance-metadata-defaults \
    --region us-east-1 \
    --http-tokens required \
    --http-put-response-hop-limit 2
```

Berikut ini adalah output contoh.

```
{
    "Return": true
}
```

Untuk melihat pengaturan akun default untuk opsi metadata instans untuk Wilayah yang ditentukan

Gunakan perintah get-instance-metadata- dedefault dan tentukan Wilayah.

```
aws ec2 get-instance-metadata-defaults --region us-east-1
```

Berikut ini adalah output contoh.

```
{
    "AccountLevel": {
        "HttpTokens": "required",
        "HttpPutResponseHopLimit": 2
    },
    "ManagedBy": "account"
}
```

Bid `ManagedBy` ang menunjukkan entitas yang mengkonfigurasi pengaturan. Dalam contoh ini, `account` menunjukkan bahwa pengaturan dikonfigurasi langsung di akun. Nilai berarti `declarative-policy` pengaturan dikonfigurasi oleh kebijakan deklaratif. Untuk informasi selengkapnya, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Untuk mengatur IMDSv2 sebagai default untuk akun untuk semua Wilayah

Gunakan perintah modify-instance-metadata-dedefault untuk mengubah pengaturan tingkat akun IMDS untuk semua Wilayah. Sertakan `--http-tokens` disetel ke `required` dan `--http-put-response-hop-limit` setel ke 2 jika instans Anda akan meng-host kontainer. Jika tidak, tent `-1` ukan untuk menunjukkan tidak ada preferensi. Ketika `-1` (tidak ada preferensi) ditentukan, saat peluncuran, nilai default adalah 2 jika AMI memiliki pengaturan`ImdsSupport: v2.0`; jika tidak maka defaultnya menjadi. 1

```
echo -e "Region          \t Modified" ; \
echo -e "--------------  \t ---------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 modify-instance-metadata-defaults \
            --region $region \
            --http-tokens required \
            --http-put-response-hop-limit 2 \
            --output text)
        echo -e "$region        \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region                  Modified
--------------          ---------
ap-south-1              True
eu-north-1              True
eu-west-3               True
...
```

Untuk melihat pengaturan akun default untuk opsi metadata instans untuk semua Wilayah

Gunakan perintah get-instance-metadata-dedefault.

```
echo -e "Region   \t Level          Hops    HttpTokens" ; \
echo -e "-------------- \t ------------   ----    ----------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 get-instance-metadata-defaults \
            --region $region \
            --output text)
        echo -e "$region \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region           Level          Hops    HttpTokens
--------------   ------------   ----    ----------
ap-south-1       ACCOUNTLEVEL   2       required
eu-north-1       ACCOUNTLEVEL   2       required
eu-west-3        ACCOUNTLEVEL   2       required
...
```

PowerShell

Untuk mengatur IMDSv2 sebagai default untuk akun untuk Wilayah yang ditentukan

Gunakan Edit-EC2InstanceMetadataDefault cmdlet dan tentukan Wilayah tempat memodifikasi pengaturan tingkat akun IMDS. Sertakan `-HttpToken` disetel ke `required` dan `-`

`HttpPutResponseHopLimit` setel ke 2 jika instans Anda akan meng-host kontainer. Jika tidak, tent `-1` ukan untuk menunjukkan tidak ada preferensi. Ketika `-1` (tidak ada preferensi) ditentukan, saat peluncuran, nilai default adalah 2 jika AMI memiliki pengaturan`ImdsSupport: v2.0`; jika tidak maka defaultnya menjadi. 1

```
Edit-EC2InstanceMetadataDefault `
    -Region us-east-1 `
    -HttpToken required `
    -HttpPutResponseHopLimit 2
```

Berikut ini adalah output contoh.

```
True
```

Untuk melihat pengaturan akun default untuk opsi metadata instans untuk Wilayah yang ditentukan

Gunakan Get-EC2InstanceMetadataDefault cmdlet dan tentukan Wilayah.

```
Get-EC2InstanceMetadataDefault -Region us-east-1 | Format-List
```

Berikut ini adalah output contoh.

```
HttpEndpoint            :
HttpPutResponseHopLimit : 2
HttpTokens              : required
InstanceMetadataTags    :
```

Untuk mengatur IMDSv2 sebagai default untuk akun untuk semua Wilayah

Gunakan Edit-EC2InstanceMetadataDefault cmdlet untuk mengubah pengaturan tingkat akun IMDS untuk semua Wilayah. Sertakan `-HttpToken` disetel ke `required` dan `-HttpPutResponseHopLimit` setel ke 2 jika instans Anda akan meng-host kontainer. Jika tidak, tent `-1` ukan untuk menunjukkan tidak ada preferensi. Ketika `-1` (tidak ada preferensi) ditentukan, saat peluncuran, nilai default adalah 2 jika AMI memiliki pengaturan`ImdsSupport: v2.0`; jika tidak maka defaultnya menjadi. 1

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
```

```
    [PSCustomObject]@{
        Region   = $_
        Modified = (Edit-EC2InstanceMetadataDefault `
                -Region $_ `
                -HttpToken required `
                -HttpPutResponseHopLimit 2)
    }
} | `
Format-Table Region, Modified -AutoSize
```

Output yang diharapkan

```
Region         Modified
------         --------
ap-south-1         True
eu-north-1         True
eu-west-3          True
...
```

Untuk melihat pengaturan akun default untuk opsi metadata instans untuk semua Wilayah

Gunakan Get-EC2InstanceMetadataDefault cmdlet.

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region = $_
        HttpPutResponseHopLimit = (Get-EC2InstanceMetadataDefault -Region
 $_).HttpPutResponseHopLimit
        HttpTokens              = (Get-EC2InstanceMetadataDefault -Region
 $_).HttpTokens
    }
} | `
Format-Table -AutoSize
```

Contoh Output

```
Region         HttpPutResponseHopLimit HttpTokens
------         ----------------------- ----------
ap-south-1                           2 required
eu-north-1                           2 required
eu-west-3                            2 required
```

```
...
```

### Menerapkan IMDSv2 di tingkat akun

Anda dapat menerapkan penggunaan IMDSv2 di tingkat akun untuk masing-masing. Wilayah AWS Saat diterapkan, instance hanya dapat diluncurkan jika dikonfigurasi untuk memerlukan IMDSv2. Penegakan ini berlaku terlepas dari bagaimana instance atau AMI dikonfigurasi.

> **Note**
>
> Sebelum mengaktifkan penegakan IMDSv2 di tingkat akun, pastikan aplikasi dan AMI Anda mendukung IMDSv2. Untuk informasi selengkapnya, lihat Jalur yang disarankan untuk mengharuskan IMDSv2. Jika penegakan IMDSv2 diaktifkan dan `httpTokens` tidak disetel ke `required` konfigurasi instans saat peluncuran, pengaturan akun, atau konfigurasi AMI, peluncuran instance akan gagal. Untuk informasi pemecahan masalah, lihat Meluncurkan IMDSv1-enabled instance gagal.

> **Note**
>
> Pengaturan ini tidak mengubah versi IMDS dari instans yang ada, tetapi memblokir mengaktifkan IMDSv1 pada instans yang saat ini menonaktifkan IMDSv1.

#### Console

Untuk menerapkan IMDSv2 untuk akun di Wilayah yang ditentukan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Untuk mengubah Wilayah AWS, gunakan pemilih Wilayah di sudut kanan atas halaman.
3. Di panel navigasi, pilih Dasbor.
4. Pada kartu Atribut Akun, di bawah Pengaturan, pilih Perlindungan dan keamanan data.
5. Di samping IMDS default, pilih Kelola.
6. Pada halaman Kelola default IMDS, lakukan hal berikut:
   a. Untuk Versi metadata, pilih V2 saja (token diperlukan).
   b. Untuk Men egakkan IMDSv2, pilih Diaktifkan.

c. Pilih Perbarui.

AWS CLI

Untuk menerapkan IMDSv2 untuk akun di Wilayah yang ditentukan

Gunakan perintah modify-instance-metadata-dedefault dan tentukan Wilayah untuk menerapkan IMDSv2.

```
aws ec2 modify-instance-metadata-defaults \
    --region us-east-1 \
    --http-tokens required \
    --http-tokens-enforced enabled
```

Berikut ini adalah output contoh.

```
{
"Return": true
}
```

Untuk melihat setelan penegakan IMDSv2 untuk akun di Wilayah tertentu

Gunakan perintah get-instance-metadata- dedefault dan tentukan Wilayah.

```
aws ec2 get-instance-metadata-defaults --region us-east-1
```

Berikut ini adalah output contoh.

```
{
    "AccountLevel": {
        "HttpTokens": "required",
        "HttpTokensEnforced": "enabled"
    },
    "ManagedBy": "account"
}
```

Bid `ManagedBy` ang menunjukkan entitas yang mengkonfigurasi pengaturan. Dalam contoh ini, `account` menunjukkan bahwa pengaturan dikonfigurasi langsung di akun. Nilai berarti `declarative-policy` pengaturan dikonfigurasi oleh kebijakan deklaratif. Untuk informasi selengkapnya, lihat Kebijakan Deklaratif di Panduan AWS Pengguna Organisasi.

Untuk menerapkan IMDSv2 untuk akun untuk semua Wilayah

Gunakan perintah modify-instance-metadata-de default untuk menerapkan IMDSv2 di semua Wilayah.

```
echo -e "Region          \t Modified" ; \
echo -e "--------------  \t ---------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 modify-instance-metadata-defaults \
            --region $region \
            --http-tokens-enforced enabled \
            --output text)
        echo -e "$region        \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region                  Modified
--------------          ---------
ap-south-1              True
eu-north-1              True
eu-west-3               True
...
```

Untuk melihat pengaturan penegakan IMDSv2 untuk akun di semua Wilayah

Gunakan perintah get-instance-metadata-dedefault.

```
echo -e "Region   \t Level           HttpTokensEnforced" ; \
echo -e "-------------- \t ------------   ----------------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
```

```
    do (output=$(
        aws ec2 get-instance-metadata-defaults \
            --region $region \
            --query 'AccountLevel.HttpTokensEnforced' \
            --output text)
        echo -e "$region \t ACCOUNTLEVEL $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region           Level          HttpTokensEnforced
--------------   ------------   ------------------
ap-south-1       ACCOUNTLEVEL   enabled
eu-north-1       ACCOUNTLEVEL   enabled
eu-west-3        ACCOUNTLEVEL   enabled
...
```

PowerShell

Untuk menerapkan IMDSv2 untuk akun di Wilayah yang ditentukan

Gunakan Edit-EC2InstanceMetadataDefault cmdlet dan tentukan Wilayah untuk menerapkan IMDSv2.

```
Edit-EC2InstanceMetadataDefault `
    -Region us-east-1 `
    -HttpToken required `
    -HttpPutResponseHopLimit 2
```

Berikut ini adalah output contoh.

```
@{
    Return = $true
}
```

Untuk melihat setelan penegakan IMDSv2 untuk akun di Wilayah tertentu

Gunakan Get-EC2InstanceMetadataDefault perintah dan tentukan Wilayah.

```
Get-EC2InstanceMetadataDefault -Region us-east-1
```

Berikut ini adalah output contoh.

```
@{
    AccountLevel = @{
        HttpTokens = "required"
        HttpTokensEnforced = "enabled"
    }
    ManagedBy = "account"
}
```

Bid `ManagedBy` ang menunjukkan entitas yang mengkonfigurasi pengaturan. Dalam contoh ini, `account` menunjukkan bahwa pengaturan dikonfigurasi langsung di akun. Nilai berarti `declarative-policy` pengaturan dikonfigurasi oleh kebijakan deklaratif. Untuk informasi selengkapnya, lihat Kebijakan Deklaratif di Panduan AWS Pengguna Organisasi.

Untuk menerapkan IMDSv2 untuk akun untuk semua Wilayah

Gunakan perintah modify-instance-metadata-de default untuk menerapkan IMDSv2 di semua Wilayah.

```
echo -e "Region          \t Modified" ; \
echo -e "--------------  \t ---------" ; \
for region in $(
    aws ec2 describe-regions \
        --region us-east-1 \
        --query "Regions[*].[RegionName]" \
        --output text
    );
    do (output=$(
        aws ec2 modify-instance-metadata-defaults \
            --region $region \
            --http-tokens-enforced enabled \
            --output text)
        echo -e "$region        \t $output"
    );
done
```

Berikut ini adalah output contoh.

```
Region                  Modified
--------------          ---------
ap-south-1              True
```

```
eu-north-1                True
eu-west-3                 True
...
```

Untuk mengatur IMDSv2 sebagai default untuk akun untuk semua Wilayah

Gunakan Edit-EC2InstanceMetadataDefault cmdlet untuk mengubah pengaturan tingkat akun IMDS untuk semua Wilayah. Sertakan `-HttpToken` disetel ke `required` dan `-HttpPutResponseHopLimit` setel ke 2 jika instans Anda akan meng-host kontainer. Jika tidak, tent `-1` ukan untuk menunjukkan tidak ada preferensi. Ketika `-1` (tidak ada preferensi) ditentukan, saat peluncuran, nilai default adalah 2 jika AMI memiliki pengaturan`ImdsSupport: v2.0`; jika tidak maka defaultnya menjadi. 1

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region   = $_
        Modified = (Edit-EC2InstanceMetadataDefault `
                -Region $_ `
                -HttpToken required `
                -HttpPutResponseHopLimit 2)
    }
} | `
Format-Table Region, Modified -AutoSize
```

Output yang diharapkan

```
Region         Modified
------         --------
ap-south-1         True
eu-north-1         True
eu-west-3          True
...
```

Untuk melihat pengaturan akun default untuk opsi metadata instans untuk semua Wilayah

Gunakan Get-EC2InstanceMetadataDefault cmdlet.

```
(Get-EC2Region).RegionName | `
    ForEach-Object {
    [PSCustomObject]@{
        Region = $_
```

```
        HttpPutResponseHopLimit = (Get-EC2InstanceMetadataDefault -Region
 $_).HttpPutResponseHopLimit
        HttpTokens              = (Get-EC2InstanceMetadataDefault -Region
 $_).HttpTokens
    }
} | `
Format-Table -AutoSize
```

Contoh Output

```
Region         HttpPutResponseHopLimit HttpTokens
------         ----------------------- ----------
ap-south-1                           2 required
eu-north-1                           2 required
eu-west-3                            2 required
...
```

Konfigurasikan instans saat peluncuran

Saat meluncurkan instans, Anda dapat mengonfigurasi instans agar memerlukan penggunaan IMDSv2 dengan mengonfigurasi bidang berikut:

- Konsol Amazon EC2: Atur Versi metadata ke V2 saja (diperlukan token).
- AWS CLI: Atur `HttpTokens` ke `required`.

Jika Anda menentukan bahwa IMDSv2 diperlukan, Anda juga harus mengaktifkan titik akhir Layanan Metadata Instans (IMDS) dengan menyetel Metadata yang dapat diakses ke Diaktifkan (konsol) atau `HttpEndpoint` ke `enabled` (AWS CLI).

Dalam lingkungan kontainer, ketika IMDSv2 diperlukan, kami sarankan mengatur batas hop ke. 2 Untuk informasi selengkapnya, lihat Pertimbangan akses metadata instans.

Console

Untuk mengharuskan penggunaan IMDSv2 pada instans baru

- Saat meluncurkan instans baru di konsol Amazon EC2, perluas Detail lanjutan, dan lakukan hal berikut:
  - Untuk Metadata yang dapat diakses, pilih Diaktifkan.

- Untuk Versi metadata, pilih V2 saja (token diperlukan).
- (Lingkungan kontainer) Untuk batas hop respons Metadata, pilih 2.

Untuk informasi selengkapnya, lihat Detail lanjutan.

AWS CLI

Untuk mengharuskan penggunaan IMDSv2 pada instans baru

Contoh run-instances berikut meluncurkan instans `c6i.large` dengan `--metadata-options` yang diatur ke `HttpTokens=required`. Jika Anda menetapkan nilai untuk `HttpTokens`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`. Karena header token aman diatur ke `required` untuk permintaan pengambilan metadata, header tersebut mengharuskan instans untuk menggunakan IMDSv2 saat meminta metadata instans.

Dalam lingkungan kontainer, ketika IMDSv2 diperlukan, kami sarankan mengatur batas hop ke with. 2 `HttpPutResponseHopLimit=2`

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c6i.large \
 ...
    --metadata-options
 "HttpEndpoint=enabled,HttpTokens=required,HttpPutResponseHopLimit=2"
```

PowerShell

Untuk mengharuskan penggunaan IMDSv2 pada instans baru

Contoh New-EC2Instance cmdlet berikut meluncurkan `c6i.large` instance dengan `MetadataOptions_HttpEndpoint` set ke `enabled` dan `MetadataOptions_HttpTokens` parameter ke. `required` Jika Anda menetapkan nilai untuk `HttpTokens`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`. Karena header token aman diatur ke `required` untuk permintaan pengambilan metadata, header tersebut mengharuskan instans untuk menggunakan IMDSv2 saat meminta metadata instans.

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType c6i.large `
```

```
    -MetadataOptions_HttpEndpoint enabled `
    -MetadataOptions_HttpTokens required
```

CloudFormation

Untuk menentukan opsi metadata untuk instance yang menggunakan CloudFormation, lihat LaunchTemplate MetadataOptions properti AWS: :EC2:: di Panduan AWS CloudFormation Pengguna.

### Konfigurasikan AMI

Saat Anda mendaftarkan AMI baru atau memodifikasi AMI yang ada, Anda dapat mengatur parameter `imds-support` ke `v2.0`. Instans yang diluncurkan dari AMI ini akan memiliki versi Metadata yang disetel ke V2 saja (token diperlukan) (konsol) atau `HttpTokens` disetel ke `required` (AWS CLI). Dengan pengaturan ini, instans mengharuskan penggunaan IMDSv2 saat meminta metadata instans.

Perhatikan bahwa jika Anda mengatur `imds-support` ke `v2.0`, instans yang diluncurkan dari AMI ini juga akan memiliki Batas hop tanggapan Metadata (konsol) atau `http-put-response-hop-limit` (AWS CLI) diatur ke 2.

> **Important**
>
> Jangan gunakan parameter ini kecuali perangkat lunak AMI Anda mendukung IMDSv2. Setelah Anda mengatur nilainya ke `v2.0`, Anda tidak dapat membatalkannya. Satu-satunya cara untuk “mengatur ulang” AMI Anda adalah dengan membuat AMI baru dari snapshot dasar.

Untuk mengonfigurasi AMI baru untuk IMDSv2

Gunakan salah satu metode berikut untuk mengkonfigurasi AMI baru untuk IMDSv2.

AWS CLI

Contoh register-image berikut mendaftarkan AMI menggunakan snapshot yang ditentukan dari volume root EBS sebagai perangkat `/dev/xvda`. Tentukan `v2.0` untuk parameter `imds-support`, sehingga instans yang diluncurkan dari AMI ini akan mengharuskan penggunaan IMDSv2 saat meminta metadata instans.

```
aws ec2 register-image \
    --name my-image \
    --root-device-name /dev/xvda \
    --block-device-mappings DeviceName=/dev/
xvda,Ebs={SnapshotId=snap-0123456789example} \
    --architecture x86_64 \
    --imds-support v2.0
```

PowerShell

Contoh Register-EC2Image cmdlet berikut mendaftarkan AMI menggunakan snapshot yang ditentukan dari volume root EBS sebagai perangkat. `/dev/xvda` Tentukan `v2.0` untuk parameter `ImdsSupport`, sehingga instans yang diluncurkan dari AMI ini akan mengharuskan penggunaan IMDSv2 saat meminta metadata instans.

```
Register-EC2Image `
    -Name 'my-image' `
    -RootDeviceName /dev/xvda `
    -BlockDeviceMapping  (
    New-Object `
        -TypeName Amazon.EC2.Model.BlockDeviceMapping `
        -Property @{
        DeviceName = '/dev/xvda';
        EBS        = (New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice -Property
 @{
                SnapshotId = 'snap-0123456789example'
                VolumeType = 'gp3'
                } )
        }  ) `
    -Architecture X86_64 `
    -ImdsSupport v2.0
```

Untuk mengonfigurasi AMI yang ada untuk IMDSv2

Gunakan salah satu metode berikut untuk mengonfigurasi AMI yang ada untuk IMDSv2.

AWS CLI

Contoh modify-image-attribute berikut memodifikasi AMI yang ada hanya untuk IMDSv2. Tentukan `v2.0` untuk parameter `imds-support`, sehingga instans yang diluncurkan dari AMI ini akan mengharuskan penggunaan IMDSv2 saat meminta metadata instans.

```
aws ec2 modify-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --imds-support v2.0
```

PowerShell

Contoh Edit-EC2ImageAttribute cmdlet berikut memodifikasi AMI yang ada hanya untuk IMDSv2. Tentukan `v2.0` untuk parameter `imds-support`, sehingga instans yang diluncurkan dari AMI ini akan mengharuskan penggunaan IMDSv2 saat meminta metadata instans.

```
Edit-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -ImdsSupport 'v2.0'
```

Gunakan kebijakan IAM

Anda dapat membuat kebijakan IAM yang melakukan salah satu hal berikut:

- Mencegah pengguna meluncurkan instans baru kecuali mereka memerlukan IMDSv2 pada instance baru.
- Mencegah pengguna memanggil ModifyInstanceMetadataOptions API untuk mengubah opsi metadata dari instance yang sedang berjalan. Batasi akses ke properti ModifyInstanceMetadataOptions HttpTokens untuk mencegah pembaruan instans yang sedang berjalan yang tidak diinginkan.
- Mencegah pengguna memanggil ModifyInstanceMetadataDefaults API untuk mengubah pengaturan default akun dari HttpTokens dan http. TokensEnforced Membatasi akses ke dua properti ini akan memastikan bahwa hanya peran resmi yang dapat mengubah default akun.

Untuk menegakkan penggunaan IMDSv2 pada semua instans baru dengan menggunakan kebijakan IAM

Untuk memastikan bahwa pengguna hanya dapat meluncurkan instans yang memerlukan penggunaan IMDSv2 saat meminta metadata instans, lakukan hal berikut:

- Batasi akses ke keduanya `ModifyInstanceMetadataOptions` dan `ModifyInstanceMetadataDefaults` API, dan lebih khusus lagi `httpTokens` `httpTokensEnforced` properti dan.

- Kemudian, atur default akun ke `httpTokens = required` dan`httpTokensEnforced = enabled`.

  Untuk contoh kebijakan IAM, lihat Cara menggunakan metadata instans.

Aktifkan titik akhir IPv4 dan IPv6 IMDS

IMDS memiliki dua titik akhir pada sebuah instance: IPv4 (`169.254.169.254`) dan IPv6 (). `[fd00:ec2::254]` Saat Anda mengaktifkan IMDS, titik akhir IPv4 secara otomatis diaktifkan. Titik akhir IPv6 tetap dinonaktifkan bahkan jika Anda meluncurkan instance ke IPv6-only subnet. Untuk mengaktifkan titik akhir IPv6, Anda harus melakukannya secara eksplisit. Saat Anda mengaktifkan titik akhir IPv6, titik akhir IPv4 tetap diaktifkan.

Anda dapat mengaktifkan titik akhir IPv6 saat peluncuran instance atau setelahnya.

Persyaratan untuk mengaktifkan titik akhir IPv6

- Jenis instance yang dipilih adalah Nitro-based instance.
- Subnet yang dipilih mendukung IPv6, di mana subnet berupa dual stack atau IPv6 saja.

Gunakan salah satu metode berikut untuk meluncurkan instance dengan titik akhir IPv6 IMDS diaktifkan.

Console

Untuk mengaktifkan titik akhir IPv6 IMDS saat peluncuran instance

- Luncurkan instans di konsol Amazon EC2 dengan menentukan hal-hal berikut ini pada Detail lanjutan:
  - Untuk titik akhir IPv6 Metadata, pilih Di aktifkan.

Untuk informasi selengkapnya, lihat Detail lanjutan.

AWS CLI

Untuk mengaktifkan titik akhir IPv6 IMDS saat peluncuran instance

Contoh run-instances berikut meluncurkan instans `c6i.large` dengan titik akhir IPv6 yang diaktifkan untuk IMDS. Untuk mengaktifkan titik akhir IPv6, pada parameter `--metadata-`

`options`, tentukan `HttpProtocolIpv6=enabled`. Jika Anda menetapkan nilai untuk `HttpProtocolIpv6`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c6i.large \
    ...
    --metadata-options "HttpEndpoint=enabled,HttpProtocolIpv6=enabled"
```

PowerShell

Untuk mengaktifkan titik akhir IPv6 IMDS saat peluncuran instance

Contoh New-EC2Instance cmdlet berikut meluncurkan `c6i.large` instance dengan titik akhir IPv6 diaktifkan untuk IMDS. Untuk mengaktifkan titik akhir IPv6, tentukan `MetadataOptions_HttpProtocolIpv6` sebagai `enabled`. Jika Anda menetapkan nilai untuk `MetadataOptions_HttpProtocolIpv6`, maka Anda juga harus mengatur `MetadataOptions_HttpEndpoint` ke `enabled`.

```
New-EC2Instance `
    -ImageId ami-0abcdef1234567890 `
    -InstanceType c6i.large `
    -MetadataOptions_HttpEndpoint enabled `
    -MetadataOptions_HttpProtocolIpv6 enabled
```

Nonaktifkan akses untuk metadata instans

Anda dapat menonaktifkan akses ke metadata instans dengan menonaktifkan IMDS saat Anda meluncurkan instans. Anda dapat mengaktifkan akses nanti dengan mengaktifkan kembali IMDS. Untuk informasi selengkapnya, lihat Aktifkan akses ke metadata instans.

**Important**

Anda dapat memilih untuk menonaktifkan IMDS saat peluncuran atau setelah peluncuran. Jika Anda menonaktifkan IMDS saat peluncuran, hal-hal berikut ini mungkin tidak berfungsi:

- Anda mungkin tidak memiliki akses SSH ke instans Anda. `public-keys/0/openssh-key`, yang merupakan kunci SSH publik instans Anda, tidak akan dapat diakses karena kunci biasanya disediakan dan diakses dari metadata instans EC2.

- Data pengguna EC2 tidak akan tersedia dan tidak akan berjalan saat instans dimulai. Data pengguna EC2 di-host di IMDS. Jika Anda menonaktifkan IMDS, Anda secara efektif mematikan akses ke data pengguna.

Untuk mengakses fungsionalitas ini, Anda dapat mengaktifkan kembali IMDS setelah peluncuran.

Console

Untuk menonaktifkan akses ke metadata instans saat peluncuran

- Luncurkan instans di konsol Amazon EC2 dengan menentukan hal-hal berikut ini pada Detail lanjutan:
  - Untuk Metadata yang dapat diakses, pilih Diaktifkan.

Untuk informasi selengkapnya, lihat Detail lanjutan.

AWS CLI

Untuk menonaktifkan akses ke metadata instans saat peluncuran

Luncurkan instans dengan `--metadata-options` diatur ke `HttpEndpoint=disabled`.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type c6i.large \
    ...
    --metadata-options "HttpEndpoint=disabled"
```

PowerShell

Untuk menonaktifkan akses ke metadata instans saat peluncuran

Contoh New-EC2Instance cmdlet berikut meluncurkan instance dengan `MetadataOptions_HttpEndpoint` set to. `disabled`

```
New-EC2Instance `
```

```
    -ImageId ami-0abcdef1234567890 `
    -InstanceType c6i.large `
    -MetadataOptions_HttpEndpoint disabled
```

CloudFormation

Untuk menentukan opsi metadata untuk instance yang menggunakan CloudFormation, lihat LaunchTemplate MetadataOptions properti AWS: :EC2:: di Panduan CloudFormation Pengguna.

Mengizinkan akses ke tanda dalam metadata instans

Secara default, tag instance tidak dapat diakses dalam metadata instance. Untuk setiap contoh, Anda harus secara eksplisit mengizinkan akses. Jika akses diizinkan, kunci tag instance harus mematuhi batasan karakter tertentu, jika tidak peluncuran instance akan gagal. Untuk informasi selengkapnya, lihat Aktifkan akses ke tag dalam metadata instance.

## Mengonfigurasi opsi metadata instans untuk instans yang ada

Anda dapat mengonfigurasi opsi metadata instans untuk instans yang ada

Anda juga dapat membuat kebijakan IAM yang mencegah pengguna mengubah opsi metadata instans pada instans yang ada. Untuk mengontrol pengguna mana yang dapat mengubah opsi metadata instance, tentukan kebijakan yang mencegah semua pengguna selain pengguna dengan peran tertentu untuk menggunakan ModifyInstanceMetadataOptions API. Untuk contoh kebijakan IAM, lihat Cara menggunakan metadata instans.

> **Note**
>
> Jika kebijakan deklaratif digunakan untuk mengonfigurasi opsi metadata instance, Anda tidak dapat memodifikasinya secara langsung di dalam akun. Untuk informasi selengkapnya, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Mengharuskan penggunaan IMDSv2

Gunakan salah satu metode berikut untuk memodifikasi opsi metadata instans pada instans yang ada untuk mengharuskan penggunaan IMDSv2 saat meminta metadata instans. Ketika IMDSv2 diharuskan, IMDSv1 tidak dapat digunakan.

> **Note**
>
> Sebelum mengharuskan IMDSv2 digunakan, pastikan instans tidak melakukan panggilan IMDSv1. Met `MetadataNoToken` CloudWatch rik melacak panggilan IMDSv1. Ketika `MetadataNoToken` merekam nol penggunaan IMDSv1 untuk sebuah instance, instance tersebut kemudian siap untuk memerlukan IMDSv2.

### Console

Untuk mengharuskan penggunaan IMDSv2 pada instans yanga ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda.
4. Pilih Tindakan, Pengaturan instans, Ubah opsi metadata instans.
5. Di kotak dialog Ubah opsi metadata instans, lakukan hal berikut:
   a. Untuk Layanan metadata instans, pilih Diaktifkan.
   b. Untuk IMDSv2, pilih Wajib
   c. Pilih Simpan.

### AWS CLI

Untuk mengharuskan penggunaan IMDSv2 pada instans yanga ada

Gunakan perintah CLI modify-instance-metadata-options dan atur parameter `http-tokens` ke `required`. Jika Anda menetapkan nilai untuk `http-tokens`, maka Anda juga harus mengatur `http-endpoint` ke `enabled`.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567890abcdef0 \
    --http-tokens required \
    --http-endpoint enabled
```

### PowerShell

Untuk mengharuskan penggunaan IMDSv2 pada instans yanga ada

Gunakan Edit-EC2InstanceMetadataOption cmdlet dan atur `HttpTokens` parameter ke. `required` Jika Anda menetapkan nilai untuk `HttpTokens`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`.

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567890abcdef0 `
    -HttpTokens required `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

Memulihkan penggunaan IMDSv1

Ketika IMDSv2 diperlukan pada instance, menggunakan permintaan IMDSv1 akan gagal. Ketika IMDSv2 bersifat opsional, maka IMDSv2 dan IMDSv1 akan berfungsi. Oleh karena itu, untuk mengembalikan IMDSv1, atur IMDSv2 ke optional (`httpTokens = optional`) menggunakan salah satu metode berikut.

Pro `httpTokensEnforced` perti IMDS juga mencegah upaya untuk mengaktifkan IMDSv1 pada instance yang ada. Saat diaktifkan untuk akun di Wilayah, upaya untuk menyetel `httpTokens` ke `optional` akan menghasilkan `Unsupported0peration` pengecualian. Untuk informasi lebih lanjut, lihatPemecahan masalah.

> **Important**
>
> Jika peluncuran instans gagal karena penegakan IMDSv2, Anda memiliki dua opsi untuk mengaktifkan peluncuran agar berhasil:
>
> - Luncurkan instance sebagai IMDSv2-only — Jika perangkat lunak yang berjalan pada instans hanya menggunakan IMDSv2 (tidak ada ketergantungan pada IMDSv1), maka Anda dapat meluncurkan instans sebagai IMDSv2 saja. Untuk melakukan ini, konfigurasikan IMDSv2 hanya dengan mengatur `httpTokens = required` parameter peluncuran atau dalam default metadata untuk akun di Wilayah.
> - Nonaktifkan penegakan — Jika perangkat lunak Anda masih bergantung pada IMDSv1, setel `httpTokensEnforced` ke `disabled` untuk akun di Wilayah. Untuk informasi selengkapnya, lihat Menerapkan IMDSv2 di tingkat akun.

Console

Untuk memulihkan penggunaan IMDSv1 pada instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda.
4. Pilih Tindakan, Pengaturan instans, Ubah opsi metadata instans.
5. Di kotak dialog Ubah opsi metadata instans, lakukan hal berikut:
   a. Untuk Layanan metadata instans, pastikan Aktifkan telah dipilih.
   b. Untuk IMDSv2, pilih Opsional.
   c. Pilih Simpan.

AWS CLI

Untuk memulihkan penggunaan IMDSv1 pada instans

Anda dapat menggunakan perintah CLI modify-instance-metadata-options dengan `http-tokens` yang diatur ke `optional` untuk memulihkan penggunaan IMDSv1 saat meminta metadata instans.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567890abcdef0 \
    --http-tokens optional \
    --http-endpoint enabled
```

PowerShell

Untuk memulihkan penggunaan IMDSv1 pada instans

Anda dapat menggunakan Edit-EC2InstanceMetadataOption cmdlet dengan `HttpTokens` set to `optional` untuk memulihkan penggunaan IMDSv1 saat meminta metadata instans.

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567890abcdef0 `
    -HttpTokens optional `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

Mengubah batas hop respons PUT

Untuk instans yang ada, Anda dapat mengubah pengaturan batas hop respons PUT.

Saat ini hanya AWS CLI AWS SDK dan yang mendukung mengubah batas hop respons PUT.

AWS CLI

Mengubah batas hop respons PUT

Gunakan perintah CLI modify-instance-metadata-options dan atur parameter `http-put-response-hop-limit` ke jumlah hop yang diperlukan. Pada instans berikut, hop limit diatur ke `3`. Perhatikan bahwa saat Anda menetapkan nilai untuk `http-put-response-hop-limit`, maka Anda juga harus mengatur `http-endpoint` ke `enabled`.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567890abcdef0 \
    --http-put-response-hop-limit 3 \
    --http-endpoint enabled
```

PowerShell

Mengubah batas hop respons PUT

Gunakan Edit-EC2InstanceMetadataOption cmdlet dan atur `HttpPutResponseHopLimit` parameter ke jumlah hop yang diperlukan. Pada instans berikut, batas hop diatur ke `3`. Perhatikan bahwa saat Anda menetapkan nilai untuk `HttpPutResponseHopLimit`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`.

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567890abcdef0 `
    -HttpPutResponseHopLimit 3 `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

Aktifkan titik akhir IPv4 dan IPv6 IMDS

IMDS memiliki dua titik akhir pada sebuah instance: IPv4 (`169.254.169.254`) dan IPv6 (). `[fd00:ec2::254]` Saat Anda mengaktifkan IMDS, titik akhir IPv4 secara otomatis diaktifkan. Titik akhir IPv6 tetap dinonaktifkan bahkan jika Anda meluncurkan instance ke IPv6-only subnet. Untuk

mengaktifkan titik akhir IPv6, Anda harus melakukannya secara eksplisit. Saat Anda mengaktifkan titik akhir IPv6, titik akhir IPv4 tetap diaktifkan.

Anda dapat mengaktifkan titik akhir IPv6 saat peluncuran instance atau setelahnya.

Persyaratan untuk mengaktifkan titik akhir IPv6

- Jenis instance yang dipilih adalah Nitro-based instance.
- Subnet yang dipilih mendukung IPv6, di mana subnet berupa dual stack atau IPv6 saja.

Saat ini hanya AWS SDK AWS CLI dan yang mendukung mengaktifkan titik akhir IPv6 IMDS setelah peluncuran instance.

AWS CLI

Untuk mengaktifkan titik akhir IPv6 IMDS untuk instans Anda

Gunakan perintah CLI modify-instance-metadata-options dan atur parameter `http-protocol-ipv6` ke `enabled`. Perhatikan bahwa saat Anda menetapkan nilai untuk `http-protocol-ipv6`, maka Anda juga harus mengatur `http-endpoint` ke `enabled`.

```
aws ec2 modify-instance-metadata-options \
 --instance-id i-1234567890abcdef0 \
 --http-protocol-ipv6 enabled \
 --http-endpoint enabled
```

PowerShell

Untuk mengaktifkan titik akhir IPv6 IMDS untuk instans Anda

Gunakan Edit-EC2InstanceMetadataOption cmdlet dan atur `HttpProtocolIpv6` parameter ke. `enabled` Perhatikan bahwa saat Anda menetapkan nilai untuk `HttpProtocolIpv6`, maka Anda juga harus mengatur `HttpEndpoint` ke `enabled`.

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567890abcdef0 `
    -HttpProtocolIpv6 enabled `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

### Aktifkan akses ke metadata instans

Anda dapat mengaktifkan akses ke metadata instans dengan mengaktifkan titik akhir HTTP IMDS pada instans Anda, apa pun versi IMDS yang Anda gunakan. Anda dapat megembalikan perubahan ini kapan saja dengan menonaktifkan titik akhir HTTP.

Gunakan salah satu metode berikut untuk menonaktifkan akses ke metadata instans pada instans.

#### Console

Untuk mengaktifkan akses ke metadata instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda.
4. Pilih Tindakan, Pengaturan instans, Ubah opsi metadata instans.
5. Di kotak dialog Ubah opsi metadata instans, lakukan hal berikut:
   a. Untuk Layanan metadata instans, pilih Diaktifkan.
   b. Pilih Simpan.

#### AWS CLI

Untuk mengaktifkan akses ke metadata instans

Gunakan perintah CLI modify-instance-metadata-options dan atur parameter `http-endpoint` ke `enabled`.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567890abcdef0 \
    --http-endpoint enabled
```

#### PowerShell

Untuk mengaktifkan akses ke metadata instans

Gunakan Edit-EC2InstanceMetadataOption cmdlet dan atur `HttpEndpoint` parameter ke. `enabled`

```
(Edit-EC2InstanceMetadataOption `
```

```
    -InstanceId i-1234567890abcdef0 `
    -HttpEndpoint enabled).InstanceMetadataOptions
```

### Nonaktifkan akses untuk metadata instans

Anda dapat menonaktifkan akses ke metadata instans dengan menonaktifkan titik akhir HTTP IMDS pada instans Anda, apa pun versi IMDS yang Anda gunakan. Anda dapat megembalikan perubahan ini kapan saja dengan mengaktifkan titik akhir HTTP.

Gunakan salah satu metode berikut untuk menonaktifkan akses ke metadata instans pada instans.

#### Console

Untuk menonaktifkan akses untuk metadata instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda.
4. Pilih Tindakan, Pengaturan instans, Ubah opsi metadata instans.
5. Di kotak dialog Ubah opsi metadata instans, lakukan hal berikut:
   a. Untuk Layanan metadata instans, hapus Aktifkan.
   b. Pilih Simpan.

#### AWS CLI

Untuk menonaktifkan akses untuk metadata instans

Gunakan perintah CLI modify-instance-metadata-options dan atur parameter `http-endpoint` ke `disabled`.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567890abcdef0 \
    --http-endpoint disabled
```

#### PowerShell

Untuk menonaktifkan akses untuk metadata instans

Gunakan Edit-EC2InstanceMetadataOption cmdlet dan atur `HttpEndpoint` parameter ke. `disabled`

```
(Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567890abcdef0 `
    -HttpEndpoint disabled).InstanceMetadataOptions
```

Mengizinkan akses ke tanda dalam metadata instans

Anda dapat mengizinkan akses ke tag dalam metadata instans pada instans yang sedang berjalan atau dihentikan. Untuk setiap contoh, Anda harus secara eksplisit mengizinkan akses. Jika akses diizinkan, kunci tag instance harus mematuhi batasan karakter tertentu, jika tidak, Anda mendapatkan kesalahan. Untuk informasi selengkapnya, lihat Aktifkan akses ke tag dalam metadata instance.

Pemecahan masalah

Memodifikasi IMDSv1-enabled instance gagal

Deskripsi

Anda mendapatkan pesan kesalahan berikut:

```
You can't launch instances with IMDSv1 because httpTokensEnforced
is enabled for this account. Either launch the instance with
httpTokens=required or contact your account owner to disable
httpTokensEnforced using the ModifyInstanceMetadataDefaults API or the
account settings in the EC2 console.
```

Penyebab

Kesalahan ini muncul saat Anda mencoba memodifikasi instance yang ada agar IMDsv1 enabled (`httpTokens = optional`) di akun di mana pengaturan akun EC2 atau kebijakan deklaratif AWS Organisasi memberlakukan penggunaan IMDSv2 (). `httpTokensEnforced = enabled`

Solusi

Jika Anda memerlukan dukungan IMDSv1 pada instans yang ada, Anda harus menonaktifkan penegakan IMDSv2 untuk akun di Wilayah. Untuk menonaktifkan penegakan IMDSv2,

setel ke. `HttpTokensEnforced` `disabled` Untuk informasi selengkapnya, lihat ModifyInstanceMetadataDefaults Referensi API Amazon EC2. Jika Anda lebih suka mengkonfigurasi pengaturan ini menggunakan konsol, lihatMenerapkan IMDSv2 di tingkat akun.

Kami menyarankan Anda menggunakan IMDSv2 saja (). `httpTokens=required` Untuk informasi selengkapnya, lihat Transisi ke penggunaan Layanan Metadata Instans Versi 2.

## Jalankan perintah saat Anda meluncurkan instance EC2 dengan input data pengguna

Saat meluncurkan instans Amazon EC2, Anda dapat meneruskan data pengguna ke instans yang digunakan untuk melakukan tugas konfigurasi otomatis, atau untuk menjalankan skrip setelah instance dimulai.

Jika Anda tertarik dengan skenario otomatisasi yang lebih kompleks, Anda dapat mempertimbangkan CloudFormation. Untuk informasi selengkapnya, lihat Menyebarkan aplikasi di Amazon EC2 dengan CloudFormation di Panduan AWS CloudFormation Pengguna.

Pada instans Linux, Anda dapat meneruskan dua jenis data pengguna ke Amazon EC2: skrip shell dan arahan cloud-init. Anda juga dapat meneruskan data ini ke wizard instance peluncuran sebagai teks biasa, sebagai file (ini berguna untuk meluncurkan instance dengan alat baris perintah), atau sebagai teks yang dikodekan base64 (untuk panggilan API).

Pada instans Windows, agen peluncuran menangani skrip data pengguna Anda.

Pertimbangan-pertimbangan

- Data pengguna diperlakukan sebagai data buram: apa yang Anda berikan adalah apa yang Anda dapatkan kembali. Terserah instans untuk menafsirkannya.
- Data pengguna harus dienkode base64. Konsol Amazon EC2 dapat melakukan pengkodean base64 untuk Anda atau menerima input yang dikodekan base64. Jika Anda mengambil data pengguna menggunakan metadata instans atau konsol, itu didekodekan base64 untuk Anda secara otomatis.
- Data pengguna dibatasi hingga 16 KB, dalam bentuk mentah, sebelum dienkode base64. Ukuran string dengan panjang n setelah enkode base64 adalah ceil(n/3)*4.
- Data pengguna adalah atribut instans. Jika Anda membuat AMI dari instans, data pengguna instans tidak disertakan dalam AMI.

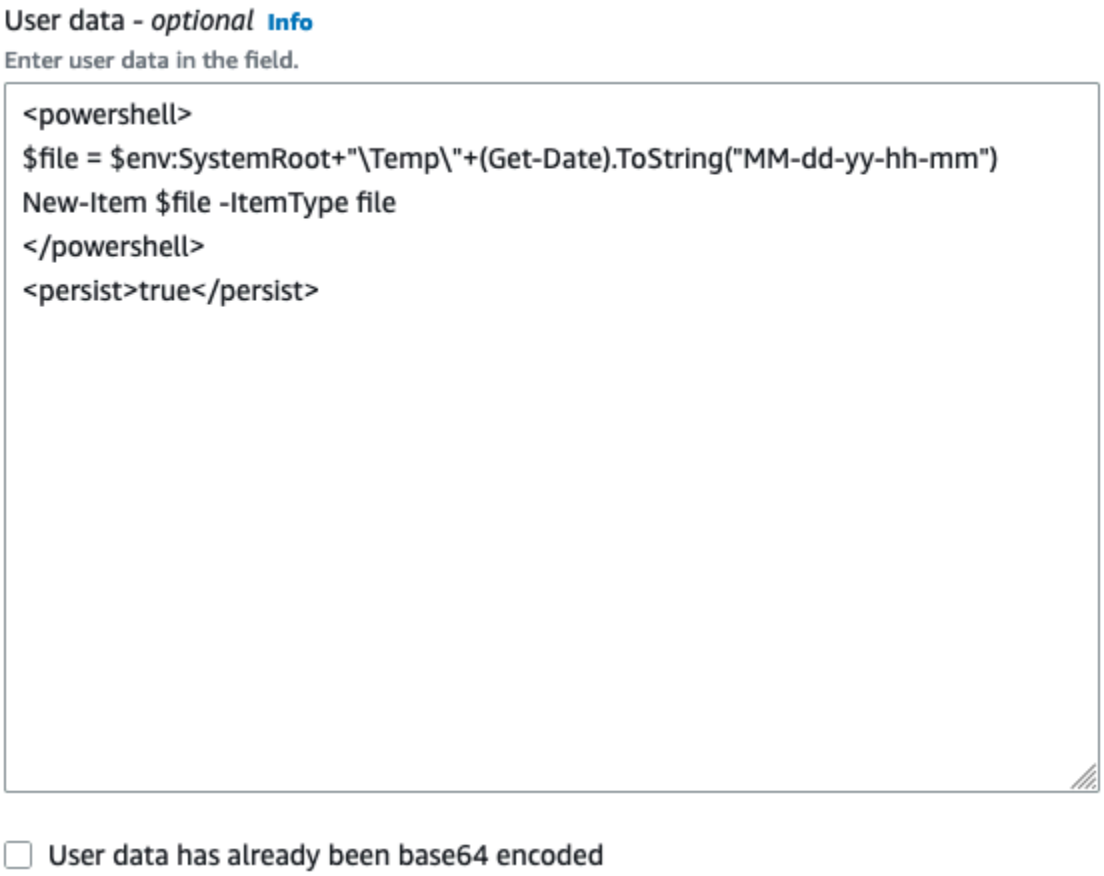
## Data pengguna di Konsol Manajemen AWS

Anda dapat menentukan data pengguna instans saat Anda meluncurkan instans. Jika volume root dari instans adalah volume EBS, Anda juga dapat menghentikan instans dan memperbarui data penggunanya.

Tentukan data pengguna instance saat diluncurkan dengan Launch Wizard

Anda dapat menentukan data pengguna saat meluncurkan instance dengan Launch Wizard di konsol EC2. Untuk menentukan data pengguna saat peluncuran, ikuti prosedur untuk meluncurkan instance. Bidang Data pengguna terletak di bagian Detail lanjutan wizard peluncuran instans. Masukkan PowerShell skrip Anda di bidang Data pengguna, lalu selesaikan prosedur peluncuran instance.

Pada tangkapan layar dari bidang Data pengguna berikut, skrip contoh membuat file di folder sementara Windows, dengan menggunakan tanggal dan waktu saat ini di nama file. Saat Anda memasukkan `<persist>true</persist>`, skrip akan dijalankan setiap kali Anda melakukan boot ulang atau memulai instans. Jika Anda membiarkan kotak centang Data pengguna sudah dikodekan base64 kosong, konsol Amazon EC2 akan melakukan pengkodean base64 untuk Anda.



Untuk informasi selengkapnya, lihat Tentukan data pengguna instance saat diluncurkan dengan Launch Wizard. Untuk contoh Linux yang menggunakan AWS CLI, lihatthe section called “Data pengguna dan AWS CLI”. Untuk contoh Windows yang menggunakan Tools untuk Windows PowerShell, lihatthe section called “Data pengguna dan Alat untuk Windows PowerShell”.

Lihat dan perbarui data pengguna instans

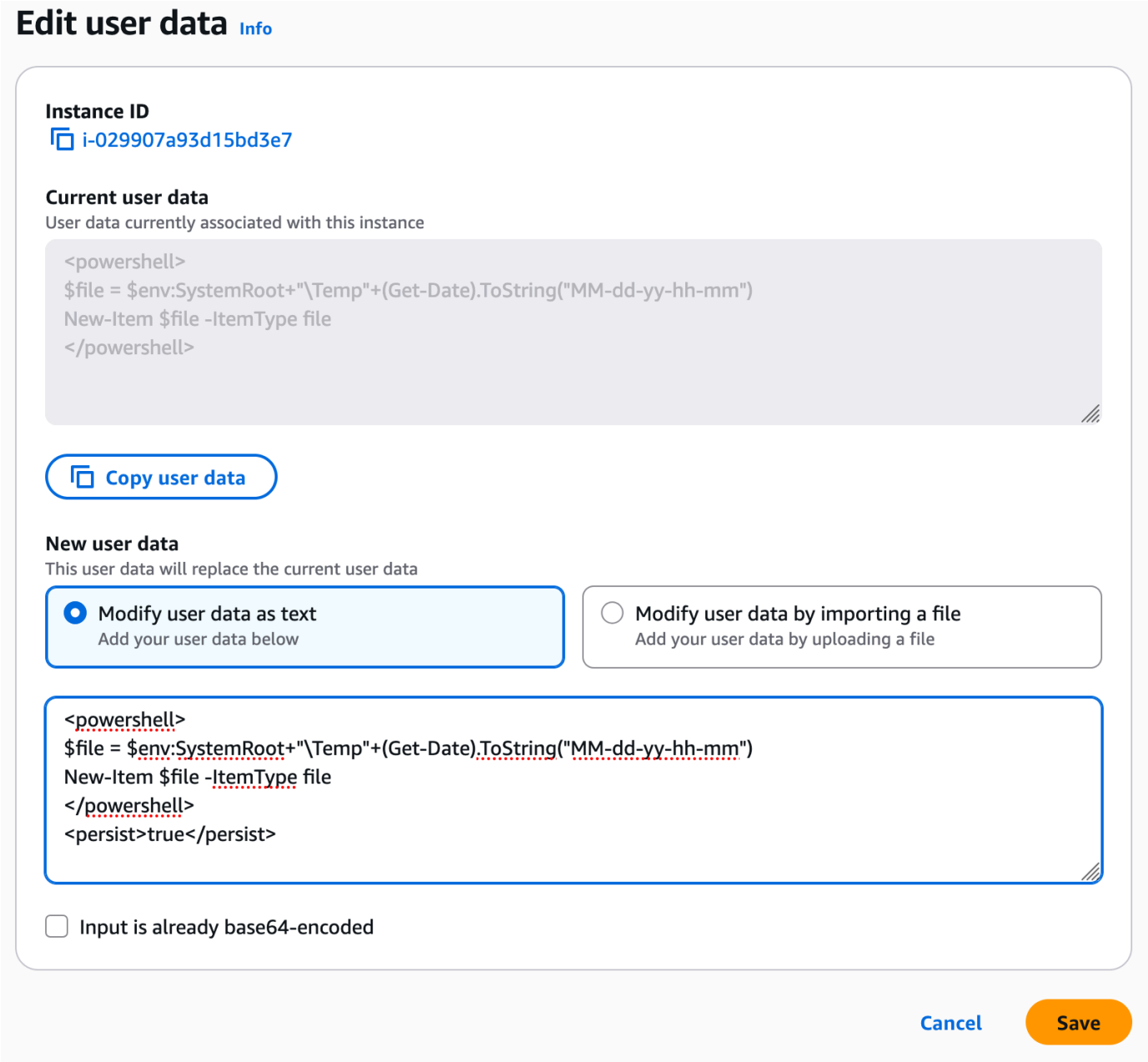
Anda dapat melihat data pengguna instans untuk semua instans, dan Anda dapat memperbarui data pengguna instans untuk instans yang dihentikan.

Untuk memperbarui data pengguna untuk sebuah instans dengan menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans dan pilih Tindakan, status instans, Hentikan instans.

> **Warning**
>
> Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

4. Ketika diminta konfirmasi, pilih Berhenti. Hal ini dapat memerlukan waktu beberapa menit sampai instans berhenti.
5. Dengan instans yang masih dipilih, pilih Tindakan, Pengaturan instans, Edit data pengguna. Anda tidak dapat mengubah data pengguna jika instans sedang berjalan, tetapi Anda dapat melihatnya.
6. Dalam kotak dialog Edit data pengguna, perbarui data pengguna, lalu pilih Simpan. Untuk menjalankan skrip data pengguna setiap kali Anda melakukan boot ulang atau memulai instans, tambahkan `<persist>true</persist>`, seperti yang ditunjukkan pada contoh berikut:




7. Mulai instans. Jika Anda mengaktifkan eksekusi data pengguna untuk boot ulang atau permulaan berikutnya, skrip data pengguna yang diperbarui akan dijalankan sebagai bagian dari proses mulai instans.

## Cara Amazon EC2 menangani data pengguna untuk instans Linux

Contoh berikut menggunakan data pengguna untuk menjalankan perintah yang mengatur server LAMP saat instance diluncurkan. Dalam setiap contoh, tugas-tugas berikut dilakukan:

- Paket perangkat lunak distribusi diperbarui.

- Server web,`php`, dan `mariadb` paket diinstal.
- `httpd`Layanan dimulai dan dihidupkan.
- Pengguna `ec2-user` ditambahkan ke grup apache.
- Kepemilikan yang sesuai dan izin file ditetapkan untuk direktori web dan file yang ada di dalamnya.
- Halaman web sederhana dibuat untuk menguji server web dan mesin PHP.

Daftar Isi


- Prasyarat
- Data pengguna dan skrip shell
- Memperbarui data pengguna instance
- Data pengguna dan arahan cloud-init
- Data pengguna dan AWS CLI
- Menggabungkan skrip shell dan arahan cloud-init


## Prasyarat

Contoh-contoh dalam topik ini mengasumsikan hal berikut:

- Instans Anda memiliki nama DNS publik yang dapat dijangkau dari internet.
- Grup keamanan yang terkait dengan instans Anda dikonfigurasi untuk mengizinkan lalu lintas SSH (port 22) sehingga Anda dapat terhubung ke instans untuk melihat file log keluaran.
- Instans Anda diluncurkan menggunakan AMI Amazon Linux. Perintah dan arahan mungkin tidak berfungsi untuk distribusi Linux lainnya. Untuk informasi selengkapnya tentang distribusi lain, seperti dukungannya untuk cloud-init, lihat dokumentasi untuk distribusi tertentu.

## Data pengguna dan skrip shell

Jika Anda terbiasa dengan skrip shell, ini adalah cara termudah dan terlengkap untuk mengirim instruksi ke sebuah instans saat peluncuran. Menambahkan tugas ini pada waktu boot akan menambah jumlah waktu yang dibutuhkan untuk booting sebuah instans. Anda harus memberikan waktu tambahan beberapa menit untuk penyelesaian tugas sebelum menguji apakah skrip pengguna telah berhasil diselesaikan.

> **Important**
>
> Secara default, skrip data pengguna dan arahan cloud-init hanya berjalan selama siklus boot saat Anda pertama kali meluncurkan sebuah instans. Anda dapat memperbarui konfigurasi untuk memastikan bahwa skrip data pengguna dan arahan cloud-init Anda berjalan setiap kali Anda memulai ulang instans. Untuk informasi selengkapnya, lihat Bagaimana cara memanfaatkan data pengguna untuk menjalankan skrip secara otomatis dengan setiap restart instans Amazon EC2 Linux saya? di Pusat Peng AWS etahuan.

Skrip shell data pengguna harus dimulai dengan `#!` karakter dan jalur ke penerjemah yang ingin Anda baca skrip (biasanya/bin/bash). Untuk pengenalan skrip shell, lihat Manual Referensi Bash di situs web Sistem Operasi GNU.

Skrip yang dimasukkan sebagai data pengguna dijalankan sebagai pengguna root pengguna, jadi jangan gunakan perintah sudo dalam skrip. Ingatlah bahwa file apa pun yang Anda buat akan menjadi milik pengguna root; jika Anda membutuhkan pengguna non-root untuk memiliki akses file, Anda harus mengubah izin yang sesuai dalam skrip. Selain itu, karena skrip tidak dijalankan secara interaktif, Anda tidak dapat menyertakan perintah yang memerlukan umpan balik pengguna (seperti yum update tanpa bendera `-y`).

Jika Anda menggunakan AWS API, termasuk AWS CLI, dalam skrip data pengguna, Anda harus menggunakan profil instance saat meluncurkan instance. Profil instance menyediakan AWS kredenSIAL yang sesuai yang diperlukan oleh skrip data pengguna untuk mengeluarkan panggilan API. Untuk informasi selengkapnya, lihat Menggunakan profil instance di Panduan Pengguna IAM. Izin yang Anda tetapkan ke peran IAM bergantung pada layanan mana yang Anda panggil dengan API. Untuk informasi selengkapnya, lihat IAM role untuk Amazon EC2.

File log keluaran cloud-init () menangkap keluaran konsol sehingga mudah untuk men-debug skrip Anda setelah peluncuran jika instans tidak berperilaku seperti yang Anda inginkan. Untuk melihat file log, hubungkan ke instans dan buka `/var/log/cloud-init-output.log`.

Ketika skrip data pengguna diproses, skrip itu akan disalin dan dijalankan dari `/var/lib/cloud/instances/`*`instance-id`*`/`. Skrip tidak dihapus setelah dijalankan. Pastikan untuk menghapus skrip data pengguna dari `/var/lib/cloud/instances/`*`instance-id`*`/` sebelum Anda membuat AMI dari instans. Jika tidak, skrip akan ada di direktori ini pada setiap instans yang diluncurkan dari AMI.

### Memperbarui data pengguna instance

Untuk memperbarui data pengguna instans, Anda harus menghentikan instans tersebut terlebih dahulu. Jika instans sedang berjalan, Anda dapat melihat data pengguna tetapi Anda tidak dapat mengubahnya.

> **Warning**
>
> Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

#### Untuk mengubah data pengguna instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans dan pilih Status instans, Hentikan instans. Jika opsi ini dinonaktifkan, instans sudah dihentikan atau volume root adalah volume penyimpanan instans.
4. Ketika diminta konfirmasi, pilih Berhenti. Hal ini dapat memerlukan waktu beberapa menit sampai instans berhenti.
5. Dengan instans yang masih dipilih, pilih Tindakan, Pengaturan instans, Edit data pengguna.
6. Ubah data pengguna sesuai kebutuhan, lalu pilih Simpan.
7. Mulai instans. Data pengguna baru terlihat di instans Anda setelah Anda memulainya; namun, skrip data pengguna tidak dijalankan.

### Data pengguna dan arahan cloud-init

Paket cloud-init mengonfigurasi aspek spesifik dari instans Amazon Linux baru saat diluncurkan; paket ini terutama mengonfigurasi file `.ssh/authorized_keys` untuk ec2-user sehingga Anda dapat masuk dengan kunci privat Anda sendiri. Untuk informasi selengkapnya tentang tugas konfigurasi yang dilakukan paket cloud-init untuk instans Amazon Linux, lihat dokumentasi berikut:

- Amazon Linux 2023 — Cloud- init yang disesuaikan
- Amazon Linux 2 — Menggunakan cloud-init di Amazon Linux 2

Arahan pengguna cloud-init dapat diteruskan ke instans saat peluncuran dengan cara yang sama seperti skrip diteruskan, meskipun sintaksnya berbeda. Untuk informasi selengkapnya tentang cloud-init, lihat https://cloudinit.readthedocs.org/en/latest/index.html.

> **Important**
>
> Secara default, skrip data pengguna dan arahan cloud-init hanya berjalan selama siklus boot saat Anda pertama kali meluncurkan sebuah instans. Anda dapat memperbarui konfigurasi untuk memastikan bahwa skrip data pengguna dan arahan cloud-init Anda berjalan setiap kali Anda memulai ulang instans. Untuk informasi selengkapnya, lihat Bagaimana cara memanfaatkan data pengguna untuk menjalankan skrip secara otomatis dengan setiap restart instans Amazon EC2 Linux saya? di Pusat Peng AWS etahuan.

Menambahkan tugas ini pada waktu boot akan menambah jumlah waktu yang dibutuhkan untuk booting sebuah instans. Anda harus memberikan waktu tambahan beberapa menit untuk penyelesaian tugas sebelum menguji apakah arahan data pengguna Anda telah selesai.

Untuk meneruskan arahan cloud-init ke instans Amazon Linux

1. Ikuti prosedur untuk meluncurkan instans. Bidang Data pengguna terletak di bagian Detail lanjutan wizard peluncuran instans. Masukkan teks arahan cloud-init Anda di bidang Data pengguna, lalu selesaikan prosedur peluncuran instans.

   Dalam contoh di bawah ini, arahan membuat dan mengkonfigurasi server web di Amazon Linux. Bar `#cloud-config` is di bagian atas diperlukan untuk mengidentifikasi perintah sebagai arahan cloud-init.

   AL2023

   ```
   #cloud-config
   package_update: true
   package_upgrade: all

   packages:
   - httpd
   - mariadb105-server
   - php8.1
   - php8.1-mysqlnd
   ```

```
runcmd:
- systemctl start httpd
- systemctl enable httpd
- [ sh, -c, "usermod -a -G apache ec2-user" ]
- [ sh, -c, "chown -R ec2-user:apache /var/www" ]
- chmod 2775 /var/www
- [ find, /var/www, -type, d, -exec, chmod, 2775, {}, \; ]
- [ find, /var/www, -type, f, -exec, chmod, 0664, {}, \; ]
- [ sh, -c, 'echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php' ]
```

AL2

```
#cloud-config
package_update: true
package_upgrade: all

packages:
- httpd
- mariadb-server

runcmd:
- [ sh, -c, "amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2" ]
- systemctl start httpd
- systemctl enable httpd
- [ sh, -c, "usermod -a -G apache ec2-user" ]
- [ sh, -c, "chown -R ec2-user:apache /var/www" ]
- chmod 2775 /var/www
- [ find, /var/www, -type, d, -exec, chmod, 2775, {}, \; ]
- [ find, /var/www, -type, f, -exec, chmod, 0664, {}, \; ]
- [ sh, -c, 'echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php' ]
```

2. Berikan waktu yang cukup bagi instans untuk meluncurkan dan menjalankan arahan di data pengguna Anda, lalu periksa untuk melihat apakah arahan Anda telah menyelesaikan tugas yang Anda maksudkan.

   Misalnya, di peramban web, masukkan URL file pengujian PHP yang dibuat arahan tersebut. URL ini adalah alamat DNS publik dari instans Anda yang diikuti dengan garis miring ke depan dan nama file.

   ```
   http://my.public.dns.amazonaws.com/phpinfo.php
   ```

Anda akan melihat halaman informasi PHP: Jika Anda tidak dapat melihat halaman informasi PHP, periksa apakah grup keamanan yang Anda gunakan berisi aturan yang mengizinkan lalu lintas HTTP (port 80). Untuk informasi selengkapnya, lihat Mengonfigurasi aturan grup keamanan.

3. (Opsional) Jika arahan Anda tidak menyelesaikan tugas yang Anda harapkan, atau jika Anda hanya ingin memverifikasi bahwa arahan Anda selesai tanpa kesalahan, hubungkan ke instans, periksa file log output (`/var/log/cloud-init-output.log`), dan cari pesan kesalahan di output. Untuk informasi debugging tambahan, Anda dapat menambahkan baris berikut ke arahan Anda:

   ```
   output : { all : '| tee -a /var/log/cloud-init-output.log' }
   ```

   Arahan ini mengirimkan output runcmd ke `/var/log/cloud-init-output.log`.

### Data pengguna dan AWS CLI

Anda dapat menggunakan AWS CLI untuk menentukan, memodifikasi, dan melihat data pengguna untuk instance Anda. Untuk informasi tentang melihat data pengguna dari instans Anda menggunakan metadata instans, lihat Mengakses metadata instans untuk instans EC2.

Di Windows, Anda dapat menggunakan al AWS Tools for Windows PowerShell ih-alih menggunakan AWS CLI. Untuk informasi selengkapnya, lihat Data pengguna dan Alat untuk Windows PowerShell.

#### Contoh: Tentukan data pengguna saat peluncuran

Untuk menentukan data pengguna saat Anda meluncurkan instans, gunakan perintah run-instances dengan parameter `--user-data`. Denganrun-instances, AWS CLI melakukan pengkodean base64 dari data pengguna untuk Anda.

Contoh berikut menunjukkan cara menentukan skrip sebagai string pada baris perintah:

```
aws ec2 run-instances --image-id ami-abcd1234 --count 1 --instance-type m3.medium \
    --key-name my-key-pair --subnet-id subnet-abcd1234 --security-group-ids sg-abcd1234
 \
    --user-data echo user data
```

Contoh berikut menunjukkan cara menentukan skrip menggunakan file teks. Pastikan untuk menggunakan awalan `file://` untuk menentukan file.

```
aws ec2 run-instances --image-id ami-abcd1234 --count 1 --instance-type m3.medium \
    --key-name my-key-pair --subnet-id subnet-abcd1234 --security-group-ids sg-abcd1234
 \
    --user-data file://my_script.txt
```

Berikut ini adalah contoh file teks dengan skrip shell.

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

Contoh: Modifikasi data pengguna dari instans yang dihentikan

Anda dapat memodifikasi data pengguna dari instans yang dihentikan menggunakan perintah modify-instance-attribute. Denganmodify-instance-attribute, AWS CLI tidak melakukan pengkodean base64 dari data pengguna untuk Anda.

- Di komputer Linux, gunakan perintah base64 untuk mengenkode data pengguna.

  ```
  base64 my_script.txt >my_script_base64.txt
  ```

- Di komputer Windows, gunakan perintah certutil untuk mengenkode data pengguna. Sebelum Anda dapat menggunakan file ini dengan AWS CLI, Anda harus menghapus baris pertama (BEGIN CERTIFICATE) dan terakhir (END CERTIFICATE).

  ```
  certutil -encode my_script.txt my_script_base64.txt
  notepad my_script_base64.txt
  ```

Menggunakan parameter `--attribute` dan `--value` untuk menggunakan file teks yang dienkode untuk menentukan data pengguna. Pastikan untuk menggunakan awalan `file://` untuk menentukan file.

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --attribute
 userData --value file://my_script_base64.txt
```

Contoh: Hapus data pengguna dari instans yang dihentikan

Untuk menghapus data pengguna yang ada, gunakan perintah modify-instance-attribute sebagai berikut:

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --user-data Value=
```

Contoh: Lihat data pengguna

Untuk mengambil data pengguna untuk sebuah instans, gunakan perintah describe-instance-attribute. Dengandescribe-instance-attribute, AWS CLI tidak melakukan decoding base64 dari data pengguna untuk Anda.

```
aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
 userData
```

Berikut ini adalah contoh output dengan data pengguna berenkode base64.

```
{
    "UserData": {
        "Value":
 "IyEvYmluL2Jhc2gKeXVtIHVwZGF0ZSAteQpzZXJ2aWNlIGh0dHBkIHN0YXJ0CmNoa2NvbmZpZyBodHRwZCBvbg=="
    },
    "InstanceId": "i-1234567890abcdef0"
}
```

- Di komputer Linux, gunakan`--query` opsi untuk mendapatkan data pengguna yang disandikan dan perintah base64 untuk mendekodekannya.

  ```
  aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
   userData --output text --query "UserData.Value" | base64 --decode
  ```

- Di komputer Windows, gunakan`--query` opsi untuk mendapatkan data pengguna yang dikodekan dan perintah certutil untuk mendekodekannya. Perhatikan bahwa output dienkode disimpan dalam suatu file dan output yang didekode disimpan di file lain.

  ```
  aws ec2 describe-instance-attribute --instance-id i-1234567890abcdef0 --attribute
   userData --output text --query "UserData.Value" >my_output.txt
  certutil -decode my_output.txt my_output_decoded.txt
  type my_output_decoded.txt
  ```

Berikut ini adalah output contoh.

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

Menggabungkan skrip shell dan arahan cloud-init

Secara default, Anda hanya dapat menyertakan satu tipe konten dalam data pengguna pada satu waktu. Namun, Anda dapat menggunakan tipe konten `text/cloud-config` dan `text/x-shellscript` dalam file bagian mime-multi untuk menyertakan skrip shell dan arahan cloud-init dalam data pengguna Anda.

Berikut ini menunjukkan format bagian mime-multi.

```
Content-Type: multipart/mixed; boundary="//"
MIME-Version: 1.0

--//
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
cloud-init directives

--//
Content-Type: text/x-shellscript; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="userdata.txt"

#!/bin/bash
shell script commands
--//--
```

Misalnya, data pengguna berikut menyertakan arahan cloud-init dan skrip bash shell. Arahan cloud-init membuat file (`/test-cloudinit/cloud-init.txt`), dan menulis `Created by cloud-init` ke file itu. Skrip bash shell membuat file (`/test-userscript/userscript.txt`) dan menulis `Created by bash shell script` ke file itu.

```
Content-Type: multipart/mixed; boundary="//"
MIME-Version: 1.0

--//
Content-Type: text/cloud-config; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="cloud-config.txt"

#cloud-config
runcmd:
- [ mkdir, /test-cloudinit ]
write_files:
- path: /test-cloudinit/cloud-init.txt
content: Created by cloud-init

--//
Content-Type: text/x-shellscript; charset="us-ascii"
MIME-Version: 1.0
Content-Transfer-Encoding: 7bit
Content-Disposition: attachment; filename="userdata.txt"

#!/bin/bash
mkdir test-userscript
touch /test-userscript/userscript.txt
echo "Created by bash shell script" >> /test-userscript/userscript.txt
--//--
```

## Cara Amazon EC2 menangani data pengguna untuk instans Windows

Pada instans Windows, agen peluncuran melakukan tugas yang terkait dengan data pengguna. Untuk informasi selengkapnya, lihat berikut ini:

- EC2Launch v2
- EC2Launch
- Layanan EC2Config

Untuk contoh perakitan `UserData` properti dalam CloudFormation template, lihat Properti Encoded Base64 dan Pro UserData perti Encoded Base64 dengan AccessKey dan. UserData SecretKey

Untuk contoh menjalankan perintah pada instance dalam grup Penskalaan Otomatis yang bekerja dengan kait siklus hidup, lihat Tutorial: Mengkonfigurasi data pengguna untuk mengambil status siklus hidup target melalui metadata instans di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

Daftar Isi


- Skrip data pengguna
- Data pengguna terkompresi
- Eksekusi data pengguna
- Data pengguna dan Alat untuk Windows PowerShell


## Skrip data pengguna

Untuk `EC2Config` atau `EC2Launch` untuk menjalankan skrip, Anda harus melampirkan skrip dalam tag khusus saat Anda menambahkannya ke data pengguna. Tag yang Anda gunakan tergantung pada apakah perintah dijalankan di jendela Command Prompt (perintah batch) atau menggunakan WindowsPowerShell.

Jika Anda menentukan skrip batch dan skri PowerShell p Windows, skrip batch berjalan terlebih dahulu dan PowerShell skrip Windows berjalan berikutnya, terlepas dari urutan kemunculannya dalam data pengguna instance.

Jika Anda menggunakan AWS API, termasuk AWS CLI, dalam skrip data pengguna, Anda harus menggunakan profil instance saat meluncurkan instance. Profil instance menyediakan AWS kredenSIAL yang sesuai yang diperlukan oleh skrip data pengguna untuk melakukan panggilan API. Untuk informasi selengkapnya, lihat Profil instans. Izin yang Anda tetapkan ke peran IAM bergantung pada layanan mana yang Anda panggil dengan API. Untuk informasi selengkapnya, lihat IAM role untuk Amazon EC2.

Tipe skrip


- Sintaks untuk skrip batch
- Sintaks untuk PowerShell skrip Windows
- Sintaks untuk skrip konfigurasi YAML
- Enkode Base64

Sintaks untuk skrip batch

Tentukan skrip batch menggunakan tanda `script`. Pisahkan perintah menggunakan jeda baris seperti yang ditunjukkan dalam contoh berikut.

```
<script>
    echo Current date and time >> %SystemRoot%\Temp\test.log
    echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
```

Secara default, skrip data pengguna dijalankan satu kali saat Anda meluncurkan instans. Untuk menjalankan skrip data pengguna setiap kali Anda melakukan boot ulang atau memulai instans, tambahkan `<persist>true</persist>` ke data pengguna.

```
<script>
    echo Current date and time >> %SystemRoot%\Temp\test.log
    echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
<persist>true</persist>
```

Agen EC2Launch v2

Untuk menjalankan skrip data pengguna XML sebagai proses terpisah dengan executeScript tugas EC2launch v2 di `UserData` tahap, tambahkan `<detach>true</detach>` ke data pengguna.

> **Note**
>
> detachTag tidak didukung oleh agen peluncuran sebelumnya.

```
<script>
    echo Current date and time >> %SystemRoot%\Temp\test.log
    echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
</script>
<detach>true</detach>
```

Sintaks untuk PowerShell skrip Windows

AMI AWS Windows menyertakan AWS Tools for Windows PowerShell, sehingga Anda dapat menentukan cmdlet ini dalam data pengguna. Jika Anda mengaitkan peran IAM dengan instans

Anda, Anda tidak perlu menentukan kredentif ke cmdlet, karena aplikasi yang berjalan pada instans menggunakan kredentif peran untuk mengakses AWS sumber daya (misalnya, bucket Amazon S3).

Tentukan PowerShell skrip Windows menggunakan `<powershell>` tag. Pisahkan perintah menggunakan jeda baris. Tag `<powershell>` tidak peka huruf besar/kecil.

Contoh:

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
```

Secara default, skrip data pengguna berjalan satu kali saat Anda meluncurkan instance. Untuk menjalankan skrip data pengguna setiap kali Anda melakukan boot ulang atau memulai instans, tambahkan `<persist>true</persist>` ke data pengguna.

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

Anda dapat menentukan satu atau lebih PowerShell argumen dengan `<powershellArguments>` tag. Jika tidak ada argumen yang diteruskan, EC2launch dan EC2launch v2 menambahkan argumen berikut secara default:. `-ExecutionPolicy Unrestricted`

Contoh:

```
<powershell>
    $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<powershellArguments>-ExecutionPolicy Unrestricted -NoProfile -NonInteractive</
powershellArguments>
```

Agen EC2Launch v2

Untuk menjalankan skrip data pengguna XML sebagai proses terpisah dengan executeScript tugas EC2launch v2 di `UserData` tahap, tambahkan `<detach>true</detach>` ke data pengguna.

> **Note**
>
> detachTag tidak didukung oleh agen peluncuran sebelumnya.

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<detach>true</detach>
```

Sintaks untuk skrip konfigurasi YAML

Jika Anda menggunakan EC2Launch v2 untuk menjalankan skrip, maka Anda dapat menggunakan format YAML. Untuk melihat tugas, detail, dan contoh konfigurasi untuk EC2Launch v2, lihat Konfigurasi EC2Launch v2.

Tentukan skrip YAML dengan tugas `executeScript`.

Contoh sintaks YAML untuk menjalankan skrip PowerShell

```
version: 1.0
tasks:
- task: executeScript
  inputs:
  - frequency: always
    type: powershell
    runAs: localSystem
    content: |-
      $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
      New-Item $file -ItemType file
```

Contoh sintaks YAML untuk menjalankan skrip batch

```
version: 1.1
tasks:
- task: executeScript
  inputs:
  - frequency: always
    type: batch
    runAs: localSystem
```

```
    content: |-
      echo Current date and time >> %SystemRoot%\Temp\test.log
      echo %DATE% %TIME% >> %SystemRoot%\Temp\test.log
```

Enkode Base64

Jika Anda menggunakan API Amazon EC2 atau alat yang tidak melakukan enkode base64 pada data pengguna, Anda harus mengenkode sendiri data pengguna. Jika tidak, kesalahan akan dicatat tentang tidak dapat menemukan tanda `script` atau `powershell` yang akan dijalankan. Berikut ini adalah contoh yang menyandikan menggunakan Windows PowerShell .

```
$UserData =
 [System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

Berikut ini adalah contoh yang menerjemahkan menggunakan PowerShell .

```
$Script =
 [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData))
```

Untuk informasi selengkapnya tentang pengkodean base64, lihat https://www.ietf.org/rfc/rfc4648.txt.

Data pengguna terkompresi

EC2launch v2 mendukung data pengguna yang di-zip sebagai metode untuk mengirimkan data pengguna yang lebih besar dari batas 16 KB yang diberlakukan oleh IMDS. Untuk menggunakan fitur ini, kompres skrip data pengguna Anda ke dalam `.zip` arsip dan berikan ke instans EC2 Anda. Ketika EC2launch v2 mendeteksi data pengguna terkompresi, secara otomatis membuka ritsleting skrip data pengguna terkompresi dan menjalankannya.

Seperti data pengguna standar, jika Anda menggunakan Amazon EC2 API atau alat yang tidak melakukan pengkodean base64 data pengguna, Anda harus menyandikan sendiri data pengguna terkompresi. Untuk informasi selengkapnya tentang batas ukuran data pengguna dan pengkodean base64, lihatMengakses metadata instans untuk instans EC2.

Eksekusi data pengguna

Secara default, semua AMI AWS Windows memiliki eksekusi data pengguna yang diaktifkan untuk peluncuran awal. Anda dapat menentukan bahwa skrip data pengguna dijalankan setiap kali instans di-boot ulang atau dimulai ulang. Atau, Anda dapat menentukan bahwa skrip data pengguna dijalankan setiap kali instans di-boot ulang atau dimulai ulang.

> **Note**
>
> Data pengguna tidak diaktifkan untuk dijalankan secara default setelah peluncuran awal. Agar data pengguna dapat dijalankan saat Anda melakukan boot ulang atau memulai instans, lihat Jalankan skrip selama reboot atau memulai berikutnya.

Skrip data pengguna dijalankan dari akun administrator lokal ketika kata sandi acak dibuat. Jika tidak, skrip data pengguna dijalankan dari akun Sistem.

Skrip peluncuran instance

Skrip dalam data pengguna instans dijalankan selama peluncuran awal instans. Jika tanda `persist` ditemukan, eksekusi data pengguna diaktifkan untuk boot ulang atau pemulaian berikutnya. File log untuk EC2Launch v2, EC2Launch, dan EC2Config berisi keluaran dari keluaran standar dan aliran kesalahan standar.

EC2Launch v2

File log untuk EC2Launch v2 adalah `C:\ProgramData\Amazon\EC2Launch\log\agent.log`.

> **Note**
>
> Itu`C:\ProgramData` folder mungkin tersembunyi. Untuk melihat folder, Anda harus menampilkan file dan folder yang tersembunyi.

Informasi berikut dicatat ketika data pengguna dijalankan:

- `Info: Converting user-data to yaml format` – Jika data pengguna disediakan dalam format XML
- `Info: Initialize user-data state` – Awal eksekusi data pengguna
- `Info: Frequency is: always` – Jika tugas data pengguna berjalan di setiap boot
- `Info: Frequency is: once` – Jika tugas data pengguna berjalan hanya sekali
- `Stage: postReadyUserData execution completed` – Akhir eksekusi data pengguna

EC2Launch

File log untuk EC2Launch adalah `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\UserdataExecution.log`.

Folder `C:\ProgramData` mungkin tersembunyi. Untuk melihat folder, Anda harus menampilkan file dan folder yang tersembunyi.

Informasi berikut dicatat ketika data pengguna dijalankan:

- `Userdata execution begins` – Awal eksekusi data pengguna
- `<persist> tag was provided: true` – Jika tanda yang masih ada ditemukan
- `Running userdata on every boot` – Jika tanda yang masih ada ditemukan
- `<powershell> tag was provided.. running powershell content` – Jika tanda powershell ditemukan
- `<script> tag was provided.. running script content` – Jika tanda skrip ditemukan
- `Message: The output from user scripts` – Jika skrip data pengguna dijalankan, maka outputnya dicatat

EC2Config

File log untuk EC2Config adalah `C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2Config.log`. Informasi berikut dicatat ketika data pengguna dijalankan:

- `Ec2HandleUserData: Message: Start running user scripts` – Awal eksekusi data pengguna
- `Ec2HandleUserData: Message: Re-enabled userdata execution` – Jika tanda yang masih ada ditemukan
- `Ec2HandleUserData: Message: Could not find <persist> and </persist>` – Jika tanda yang masih ada tidak ditemukan
- `Ec2HandleUserData: Message: The output from user scripts` – Jika skrip data pengguna dijalankan, maka outputnya dicatat

Jalankan skrip selama reboot atau memulai berikutnya

Saat Anda memperbarui data pengguna instans, konten data pengguna yang diperbarui secara otomatis tercermin dalam metadata instans saat Anda melakukan reboot atau memulai instance berikutnya. Namun, tergantung pada agen peluncuran yang diinstal, konfigurasi tambahan mungkin

diperlukan untuk mengonfigurasi skrip data pengguna agar dijalankan pada reboot atau mulai berikutnya.

Jika Anda memilih opsi Shutdown with Sysprep, skrip data pengguna akan berjalan saat instance dimulai atau diboot ulang berikutnya, bahkan jika Anda tidak mengaktifkan eksekusi data pengguna untuk reboot atau memulai berikutnya.

Untuk petunjuk untuk mengaktifkan eksekusi data pengguna, pilih tab yang cocok dengan agen peluncuran Anda.

EC2Launch v2

Tidak seperti EC2launch v1, EC2launch v2 mengevaluasi tugas data pengguna pada setiap boot. Tidak perlu menjadwalkan tugas data pengguna secara manual. Data pengguna berjalan berdasarkan frekuensi yang disertakan atau opsi bertahan.

Untuk skrip data pengguna XML

Untuk menjalankan skrip data pengguna pada setiap boot, tambahkan `<persist>true</persist>` flag ke data pengguna. Jika flag persistent tidak disertakan, skrip data pengguna hanya berjalan pada boot awal.

Untuk data pengguna YAML

- Untuk menjalankan tugas dalam data pengguna pada boot awal, atur tugas `frequency` ke`once`.
- Untuk menjalankan tugas di data pengguna pada setiap boot, atur tugas `frequency` ke`always`.

EC2Launch

1. Hubungkan ke instans Windows Anda.
2. Buka jendela PowerShell perintah dan jalankan salah satu perintah berikut:

   Jalankan sekali

   Untuk menjalankan data pengguna sekali pada boot berikutnya, gunakan `-Schedule` flag.

   ```
   C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
   Schedule
   ```

Jalankan pada semua sepatu bot berikutnya

Untuk menjalankan data pengguna pada semua boot berikutnya, gunakan ben `-SchedulePerBoot` dera.

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
SchedulePerBoot
```

3. Putuskan sambungan dari instans Windows Anda. Untuk menjalankan skrip yang diperbarui saat instans dimulai lagi nanti, hentikan instans dan perbarui data pengguna.

EC2Config

1. Hubungkan ke instans Windows Anda.
2. Buka `C:\Program Files\Amazon\Ec2ConfigService\Ec2ConfigServiceSetting.exe`.
3. Untuk Data Pengguna , pilih Aktifkan UserData eksekusi untuk layanan berikutnya dimulai .
4. Putuskan sambungan dari instans Windows Anda. Untuk menjalankan skrip yang diperbarui saat instans dimulai lagi nanti, hentikan instans dan perbarui data pengguna.

Data pengguna dan Alat untuk Windows PowerShell

Anda dapat menggunakan Tools for Windows PowerShell untuk menentukan, memodifikasi, dan melihat data pengguna untuk instance Anda. Untuk informasi tentang melihat data pengguna dari instans Anda menggunakan metadata instans, lihat Mengakses metadata instans untuk instans EC2. Untuk informasi tentang data pengguna dan AWS CLI, lihatData pengguna dan AWS CLI.

Contoh: Tentukan data pengguna instans saat peluncuran

Buat file teks dengan data pengguna instans. Untuk menjalankan skrip data pengguna setiap kali Anda melakukan boot ulang atau memulai instans, tambahkan `<persist>true</persist>`, seperti yang ditunjukkan pada contoh berikut.

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
```

```
<persist>true</persist>
```

Untuk menentukan data pengguna instance saat Anda meluncurkan instance, gunakan New-EC2Instance perintah. Perintah ini tidak melakukan enkode base64 pada data pengguna untuk Anda. Gunakan perintah berikut untuk menyandikan data pengguna dalam file teks bernama.`script.txt`

```
PS C:\> $Script = Get-Content -Raw script.txt
PS C:\> $UserData =
 [System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Script))
```

Gunakan parameter `-UserData` untuk meneruskan data pengguna ke perintah New-EC2Instance.

```
PS C:\> New-EC2Instance -ImageId ami-abcd1234 -MinCount 1 -MaxCount 1 -
InstanceType m3.medium \
    -KeyName my-key-pair -SubnetId subnet-12345678 -SecurityGroupIds sg-1a2b3c4d \
    -UserData $UserData
```

Contoh: Perbarui data pengguna instans untuk instans yang dihentikan

Anda dapat memodifikasi data pengguna dari instance yang dihentikan menggunakan Edit-EC2InstanceAttribute perintah.

Buat file teks dengan skrip baru. Gunakan perintah berikut untuk menyandikan data pengguna dalam file teks bernama.`new-script.txt`

```
PS C:\> $NewScript = Get-Content -Raw new-script.txt
PS C:\> $NewUserData =
 [System.Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($NewScript))
```

Gunakan parameter `-UserData` dan `-Value` untuk menentukan data pengguna.

```
PS C:\> Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute userData -
Value $NewUserData
```

Contoh: Lihat data pengguna instans

Untuk mengambil data pengguna untuk sebuah instance, gunakan Get-EC2InstanceAttribute perintah.

```
PS C:\> (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -Attribute
 userData).UserData
```

Berikut ini adalah output contoh. Perhatikan bahwa data pengguna dienkode.

```
PHBvd2Vyc2hlbGw
+DQpSZW5hbWUtQ29tcHV0ZXIgLU5ld05hbWUgdXNlci1kYXRhLXRlc3QNCjwvcG93ZXJzaGVsbD4=
```

Gunakan perintah berikut untuk menyimpan data pengguna yang dienkode dalam variabel dan kemudian mendekodekannya.

```
PS C:\> $UserData_encoded = (Get-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -
Attribute userData).UserData
PS C:
\> [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($UserData_encoded)
```

Berikut ini adalah contoh output.

```
<powershell>
    $file = $env:SystemRoot + "\Temp\" + (Get-Date).ToString("MM-dd-yy-hh-mm")
    New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

Contoh: Ubah nama instans agar sesuai dengan nilai tanda

Anda dapat menggunakan Get-EC2Tag perintah untuk membaca nilai tag, mengganti nama instance pada boot pertama agar sesuai dengan nilai tag, dan reboot. Untuk menjalankan perintah ini dengan sukses, Anda harus memiliki peran dengan izin `ec2:DescribeTags` yang dilampirkan ke instans karena informasi tag diambil dengan panggilan API. Untuk informasi selengkapnya tentang izin pengaturan dengan menggunakan peran IAM, lihatMelampirkan IAM role ke instans.

IMDSv2

```
<powershell>
    [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri 'http://169.254.169.254/latest/api/token' -
UseBasicParsing
    $instanceId = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" =
 $token} -Method GET -Uri 'http://169.254.169.254/latest/meta-data/instance-id' -
UseBasicParsing
```

```
 $nameValue = (Get-EC2Tag -Filter @{Name="resource-id";Value=
$instanceid},@{Name="key";Value="Name"}).Value
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

IMDSv1

```
<powershell>
 $instanceId = (Invoke-WebRequest http://169.254.169.254/latest/meta-data/instance-
id -UseBasicParsing).content
 $nameValue = (Get-EC2Tag -Filter @{Name="resource-id";Value=
$instanceid},@{Name="key";Value="Name"}).Value
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

Anda juga dapat mengganti nama instans menggunakan tanda dalam metadata instans, jika instans Anda dikonfigurasi untuk mengakses tanda dari metadata instans. Lihat informasi yang lebih lengkap di Melihat tag untuk instans EC2 menggunakan metadata instans.

IMDSv2

```
<powershell>
```

```
    [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} -Method PUT -Uri 'http://169.254.169.254/latest/api/token' -
UseBasicParsing
 $nameValue = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token}
 -Method GET -Uri 'http://169.254.169.254/latest/meta-data/tags/instance/Name' -
UseBasicParsing
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

IMDSv1

```
<powershell>
 $nameValue = Get-EC2InstanceMetadata -Path /tags/instance/Name
 $pattern = "^(?![0-9]{1,15}$)[a-zA-Z0-9-]{1,15}$"
 #Verify Name Value satisfies best practices for Windows hostnames
 If ($nameValue -match $pattern)
     {Try
         {Rename-Computer -NewName $nameValue -Restart -ErrorAction Stop}
     Catch
         {$ErrorMessage = $_.Exception.Message
         Write-Output "Rename failed: $ErrorMessage"}}
 Else
     {Throw "Provided name not a valid hostname. Please ensure Name value is between
 1 and 15 characters in length and contains only alphanumeric or hyphen characters"}
</powershell>
```

## Identifikasi setiap instance yang diluncurkan dalam satu permintaan

Contoh ini menunjukkan bagaimana Anda dapat menggunakan data pengguna dan metadata instans untuk mengonfigurasi instans Amazon EC2 Anda.

> **Note**
>
> Contoh di bagian ini menggunakan alamat IPv4 IMDS: `169.254.169.254`. Jika Anda mengambil metadata instans untuk instans EC2 melalui alamat IPv6, pastikan Anda mengaktifkan dan menggunakan alamat IPv6 sebagai gantinya: `[fd00:ec2::254]` Alamat IPv6 IMDS kompatibel dengan perintah IMDSv2. Alamat IPv6 hanya dapat diakses pada Nitro-based instance di IPv6-supported subnet (dual stack atau IPv6 saja).

Alice ingin meluncurkan empat instans AMI basis data favoritnya, di mana instans pertama bertindak sebagai instans asli dan tiga instans lainnya bertindak sebagai replika. Saat meluncurkannya, dia ingin menambahkan data pengguna tentang strategi replikasi untuk setiap replika. Dia sadar bahwa data ini akan tersedia untuk keempat instans, jadi dia perlu menyusun data pengguna dengan cara yang memungkinkan setiap instans untuk mengenali bagian mana yang dapat diterapkan padanya. Dia bisa melakukan ini dengan menggunakan nilai metadata instans `ami-launch-index`, yang akan unik untuk setiap instans. Jika ia memulai lebih dari satu instans secara bersamaan, maka `ami-launch-index` menunjukkan urutan peluncuran instans tersebut. Nilai instans pertama yang diluncurkan adalah `0`.

Berikut adalah data pengguna yang telah dibuat oleh Alice.

```
replicate-every=1min | replicate-every=5min | replicate-every=10min
```

Data `replicate-every=1min` menentukan konfigurasi replika pertama, `replicate-every=5min` menentukan konfigurasi replika kedua, dan seterusnya. Alice memutuskan untuk memberikan data ini sebagai string ASCII dengan simbol pipa (`|`) yang membatasi data untuk instans terpisah.

Alice meluncurkan empat instans menggunakan perintah run-instances, dengan menentukan data pengguna.

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --count 4 \
    --instance-type t2.micro \
    --user-data "replicate-every=1min | replicate-every=5min | replicate-every=10min"
```

Setelah diluncurkan, semua instans memiliki salinan data pengguna dan metadata umum yang ditampilkan di sini:

- ID AMI: ami-0abcdef1234567890
- ID Reservasi: r-1234567890abcabc0
- Kunci publik: tidak ada
- Nama grup keamanan: default
- Tipe instans: t2.micro

Namun, setiap instance memiliki metadata unik, seperti yang ditunjukkan pada tabel berikut.

| Metadata | Nilai |
|---|---|
| instance-id | i-1234567890abcdef0 |
| ami-launch-index | 0 |
| public-hostname | ec2-203-0-113-25.compute-1.amazonaws.com |
| public-ipv4 | 67.202.51.223 |
| local-hostname | ip-10-251-50-12.ec2.internal |
| local-ipv4 | 10.251.50.35 |

| Metadata | Nilai |
|---|---|
| instance-id | i-0598c7d356eba48d7 |
| ami-launch-index | 1 |
| public-hostname | ec2-67-202-51-224.compute-1.amazonaws.com |
| public-ipv4 | 67.202.51.224 |
| local-hostname | ip-10-251-50-36.ec2.internal |
| local-ipv4 | 10.251.50.36 |

| Metadata | Nilai |
| --- | --- |
| instance-id | i-0ee992212549ce0e7 |
| ami-launch-index | 2 |
| public-hostname | ec2-67-202-51-225.compute-1.amazonaws.com |
| public-ipv4 | 67.202.51.225 |
| local-hostname | ip-10-251-50-37.ec2.internal |
| local-ipv4 | 10.251.50.37 |

| Metadata | Nilai |
| --- | --- |
| instance-id | i-1234567890abcdef0 |
| ami-launch-index | 3 |
| public-hostname | ec2-67-202-51-226.compute-1.amazonaws.com |
| public-ipv4 | 67.202.51.226 |
| local-hostname | ip-10-251-50-38.ec2.internal |
| local-ipv4 | 10.251.50.38 |

Alice bisa menggunakan nilai `ami-launch-index` untuk menentukan bagian mana dari data pengguna yang berlaku untuk instans tertentu.

1. Dia terhubung salah satu instans, dan mengambil `ami-launch-index` untuk instans tersebut agar dapat memastikan bahwa instans itu adalah salah satu replika:

   IMDSv2

   ```
   [ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/meta-data/api/
   token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
   ```

```
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/ami-launch-index
2
```

Untuk langkah-langkah berikut, permintaan IMDSv2 menggunakan token yang disimpan dari perintah IMDSv2 sebelumnya, dengan asumsi token belum kedaluwarsa.

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/ami-launch-index
2
```

2. Dia menyimpan `ami-launch-index` sebagai variabel.

   IMDSv2

   ```
   [ec2-user ~]$ ami_launch_index=`curl -H "X-aws-ec2-metadata-token: $TOKEN"
    http://169.254.169.254/latest/meta-data/ami-launch-index`
   ```

   IMDSv1

   ```
   [ec2-user ~]$ ami_launch_index=`curl http://169.254.169.254/latest/meta-data/ami-
   launch-index`
   ```

3. Dia menyimpan data pengguna sebagai variabel.

   IMDSv2

   ```
   [ec2-user ~]$ user_data=`curl -H "X-aws-ec2-metadata-token: $TOKEN"
    http://169.254.169.254/latest/user-data`
   ```

   IMDSv1

   ```
   [ec2-user ~]$ user_data=`curl http://169.254.169.254/latest/user-data`
   ```

4. Akhirnya, Alice menggunakan perintah cut untuk mengekstrak bagian dari data pengguna yang berlaku untuk instans itu.

   IMDSv2

   ```
   [ec2-user ~]$ echo $user_data | cut -d"|" -f"$ami_launch_index"
   replicate-every=5min
   ```

IMDSv1

```
[ec2-user ~]$ echo $user_data | cut -d"|" -f"$ami_launch_index"
replicate-every=5min
```

# Mendeteksi apakah host adalah instance EC2

Anda mungkin perlu mengetahui apakah aplikasi atau situs web Anda berjalan pada instance EC2, terutama jika Anda memiliki lingkungan komputasi campuran. Anda dapat menggunakan salah satu opsi berikut untuk menentukan apakah host untuk aplikasi atau situs web Anda adalah instans EC2.

Opsi

- Memeriksa dokumen identitas instans
- Periksa sistem UUID
- Periksa pengenal pembuatan mesin virtual sistem

## Memeriksa dokumen identitas instans

Setiap instance memiliki dokumen identitas instance yang ditandatangani yang dapat Anda verifikasi secara kriptografis. Anda dapat menemukan dokumen-dokumen ini menggunakan Instance Metadata Service (IMDS).

Untuk informasi selengkapnya, lihat Dokumen identitas instans.

## Periksa sistem UUID

Anda bisa mendapatkan UUID sistem dan mencari di oktet awal UUID untuk `EC2` (di Linux, ini mungkin huruf kecil). `ec2` Metode ini cepat, tetapi berpotensi tidak akurat karena ada kemungkinan kecil bahwa sistem yang bukan instance EC2 dapat memiliki UUID yang dimulai dengan karakter ini. Selain itu, beberapa versi SMBIOS menggunakan format little-endian, yang tidak termasuk `EC2` di awal UUID. Ini mungkin terjadi untuk instans EC2 yang menggunakan SMBIOS 2.4 untuk Windows, atau untuk distribusi Linux selain Amazon Linux yang memiliki implementasi SMBIOS sendiri.

Contoh Linux: Dapatkan UUID dari DMI (hanya HVM AMI)

Gunakan perintah berikut untuk mendapatkan UUID menggunakan Desktop Management Interface (DMI):

```
[ec2-user ~]$ sudo dmidecode --string system-uuid
```

Dalam output contoh berikut, UUID dimulai dengan "EC2", yang menunjukkan bahwa sistem tersebut mungkin adalah instans EC2.

```
EC2E1916-9099-7CAF-FD21-012345ABCDEF
```

Dalam contoh output berikut, UUID direpresentasikan dalam format little-endian.

```
45E12AEC-DCD1-B213-94ED-012345ABCDEF
```

Atau, untuk instans yang dibangun di sistem Nitro, Anda dapat menggunakan perintah berikut:

```
[ec2-user ~]$ cat /sys/devices/virtual/dmi/id/board_asset_tag
```

Jika output-nya adalah ID instans, seperti output contoh berikut, maka sistemnya adalah instans EC2:

```
i-0af01c0123456789a
```

Contoh Linux: Dapatkan UUID dari hypervisor (hanya PV AMI)

Gunakan perintah berikut untuk mendapatkan UUID dari hypervisor:

```
[ec2-user ~]$ cat /sys/hypervisor/uuid
```

Dalam output contoh berikut, UUID dimulai dengan "ec2", yang menunjukkan bahwa sistem tersebut mungkin adalah instans EC2.

```
ec2e1916-9099-7caf-fd21-012345abcdef
```

Contoh Windows: Dapatkan UUID menggunakan WMI atau Windows PowerShell

Gunakan baris perintah Instrumentasi Manajemen Windows (WMIC) sebagai berikut:

```
wmic path win32_computersystemproduct get uuid
```

Atau, jika Anda menggunakan Windows PowerShell, gunakan Get-WmiObject cmdlet sebagai berikut:

```
PS C:\> Get-WmiObject -query "select uuid from Win32_ComputerSystemProduct" | Select
 UUID
```

Dalam output contoh berikut, UUID dimulai dengan "EC2", yang menunjukkan bahwa sistem tersebut mungkin adalah instans EC2.

```
EC2AE145-D1DC-13B2-94ED-012345ABCDEF
```

Untuk instans yang menggunakan SMBIOS 2.4, UUID mungkin direpresentasikan dalam format little-endian; misalnya:

```
45E12AEC-DCD1-B213-94ED-012345ABCDEF
```

## Periksa pengenal pembuatan mesin virtual sistem

Pengidentifikasi generasi mesin virtual terdiri dari buffer unik 128-bit yang diartikan sebagai pengidentifikasi integer acak kriptografi. Anda dapat mengambil pengenal pembuatan mesin virtual untuk mengidentifikasi instans Amazon Elastic Compute Cloud Anda. Pengidentifikasi generasi diekspos dalam sistem operasi tamu instans melalui entri tabel ACPI. Nilai akan berubah jika mesin Anda dikloning, disalin, atau diimpor ke AWS, seperti dengan VM Import/Export.

Contoh: Ambil pengidentifikasi pembuatan mesin virtual dari Linux

Anda dapat menggunakan perintah berikut untuk mengambil pengenal pembuatan mesin virtual dari instans Anda yang menjalankan Linux.

Amazon Linux 2

1. Perbarui paket perangkat lunak yang ada, jika perlu, menggunakan perintah berikut:

   ```
   sudo yum update
   ```

2. Jika perlu, sumber paket busybox dengan perintah berikut ini:

   ```
   sudo curl https://www.rpmfind.net/linux/epel/next/8/Everything/x86_64/Packages/
   b/busybox-1.35.0-2.el8.next.x86_64.rpm --output busybox.rpm
   ```

3. Jika perlu, instal paket prasyarat menggunakan perintah berikut:

```
sudo yum install busybox.rpm iasl -y
```

4. Jalankan `iasl` perintah berikut untuk menghasilkan output dari tabel ACPI:

```
sudo iasl -p ./SSDT2 -d /sys/firmware/acpi/tables/SSDT2
```

5. Jalankan perintah berikut untuk meninjau output dari `iasl` perintah:

```
cat SSDT2.dsl
```

Output harus menghasilkan ruang alamat yang diperlukan untuk mengambil pengenal pembuatan mesin virtual:

```
Intel ACPI Component Architecture
ASL+ Optimizing Compiler/Disassembler version 20190509
Copyright (c) 2000 - 2019 Intel Corporation

File appears to be binary: found 32 non-ASCII characters, disassembling
Binary file appears to be a valid ACPI table, disassembling
Input file /sys/firmware/acpi/tables/SSDT2, Length 0x7B (123) bytes
ACPI: SSDT 0x0000000000000000 00007B (v01 AMAZON AMZNSSDT 00000001 AMZN
 00000001)
Pass 1 parse of [SSDT]
Pass 2 parse of [SSDT]
Parsing Deferred Opcodes (Methods/Buffers/Packages/Regions)

Parsing completed
Disassembly completed
ASL Output:    ./SSDT2.dsl - 1065 bytes
$
/*
* Intel ACPI Component Architecture
* AML/ASL+ Disassembler version 20190509 (64-bit version)
* Copyright (c) 2000 - 2019 Intel Corporation
*
* Disassembling to symbolic ASL+ operators
*
* Disassembly of /sys/firmware/acpi/tables/SSDT2, Tue Mar 29 16:15:14 2022
*
* Original Table Header:
*     Signature        "SSDT"
```

```
*     Length           0x0000007B (123)
*     Revision         0x01
*     Checksum         0xB8
*     OEM ID           "AMAZON"
*     OEM Table ID     "AMZNSSDT"
*     OEM Revision     0x00000001 (1)
*     Compiler ID      "AMZN"
*     Compiler Version 0x00000001 (1)
*/
DefinitionBlock ("", "SSDT", 1, "AMAZON", "AMZNSSDT", 0x00000001)
{
Scope (\_SB)
{
    Device (VMGN)
    {
        Name (_CID, "VM_Gen_Counter")  // _CID: Compatible ID
        Name (_DDN, "VM_Gen_Counter")  // _DDN: DOS Device Name
        Name (_HID, "AMZN0000")  // _HID: Hardware ID
        Name (ADDR, Package (0x02)
        {
            0xFED01000,
            Zero
        })
    }
}
}
```

6. (Opsional) Tingkatkan izin terminal Anda untuk langkah-langkah yang tersisa dengan perintah berikut:

```
sudo -s
```

7. Gunakan perintah berikut untuk menyimpan ruang alamat yang dikumpulkan sebelumnya:

```
VMGN_ADDR=0xFED01000
```

8. Gunakan perintah berikut untuk melakukan iterasi melalui ruang alamat dan membuat pengidentifikasi pembuatan mesin virtual:

```
for offset in 0x0 0x4 0x8 0xc; do busybox devmem $(($VMGN_ADDR + $offset)) | sed
 's/0x//' | sed -z '$ s/\n$//' >> vmgenid; done
```

9. Ambil pengidentifikasi pembuatan mesin virtual dari file output dengan perintah berikut:

```
cat vmgenid ; echo
```

Output Anda harus serupa dengan berikut:

```
EC2F335D979132C4165896753E72BD1C
```

Ubuntu

1. Perbarui paket perangkat lunak yang ada, jika perlu, menggunakan perintah berikut:

```
sudo apt update
```

2. Jika perlu, instal paket prasyarat menggunakan perintah berikut:

```
sudo apt install busybox iasl -y
```

3. Jalankan `iasl` perintah berikut untuk menghasilkan output dari tabel ACPI:

```
sudo iasl -p ./SSDT2 -d /sys/firmware/acpi/tables/SSDT2
```

4. Jalankan perintah berikut untuk meninjau output dari `iasl` perintah:

```
cat SSDT2.dsl
```

Output harus menghasilkan ruang alamat yang diperlukan untuk mengambil pengenal pembuatan mesin virtual:

```
Intel ACPI Component Architecture
ASL+ Optimizing Compiler/Disassembler version 20190509
Copyright (c) 2000 - 2019 Intel Corporation

File appears to be binary: found 32 non-ASCII characters, disassembling
Binary file appears to be a valid ACPI table, disassembling
Input file /sys/firmware/acpi/tables/SSDT2, Length 0x7B (123) bytes
ACPI: SSDT 0x0000000000000000 00007B (v01 AMAZON AMZNSSDT 00000001 AMZN
 00000001)
Pass 1 parse of [SSDT]
Pass 2 parse of [SSDT]
Parsing Deferred Opcodes (Methods/Buffers/Packages/Regions)
```

```
Parsing completed
Disassembly completed
ASL Output:    ./SSDT2.dsl - 1065 bytes
$
/*
* Intel ACPI Component Architecture
* AML/ASL+ Disassembler version 20190509 (64-bit version)
* Copyright (c) 2000 - 2019 Intel Corporation
*
* Disassembling to symbolic ASL+ operators
*
* Disassembly of /sys/firmware/acpi/tables/SSDT2, Tue Mar 29 16:15:14 2022
*
* Original Table Header:
*     Signature        "SSDT"
*     Length           0x0000007B (123)
*     Revision         0x01
*     Checksum         0xB8
*     OEM ID           "AMAZON"
*     OEM Table ID     "AMZNSSDT"
*     OEM Revision     0x00000001 (1)
*     Compiler ID      "AMZN"
*     Compiler Version 0x00000001 (1)
*/
DefinitionBlock ("", "SSDT", 1, "AMAZON", "AMZNSSDT", 0x00000001)
{
Scope (\_SB)
{
    Device (VMGN)
    {
        Name (_CID, "VM_Gen_Counter")  // _CID: Compatible ID
        Name (_DDN, "VM_Gen_Counter")  // _DDN: DOS Device Name
        Name (_HID, "AMZN0000")  // _HID: Hardware ID
        Name (ADDR, Package (0x02)
        {
            0xFED01000,
            Zero
        })
    }
}
}
```

5. (Opsional) Tingkatkan izin terminal Anda untuk langkah-langkah yang tersisa dengan perintah berikut:

```
sudo -s
```

6. Gunakan perintah berikut untuk menyimpan ruang alamat yang dikumpulkan sebelumnya:

```
VMGN_ADDR=0xFED01000
```

7. Gunakan perintah berikut untuk melakukan iterasi melalui ruang alamat dan membuat pengidentifikasi pembuatan mesin virtual:

```
for offset in 0x0 0x4 0x8 0xc; do busybox devmem $(($VMGN_ADDR + $offset)) | sed
 's/0x//' | sed -z '$ s/\n$//' >> vmgenid; done
```

8. Ambil pengidentifikasi pembuatan mesin virtual dari file output dengan perintah berikut:

```
cat vmgenid ; echo
```

   Output Anda harus serupa dengan berikut:

```
EC2F335D979132C4165896753E72BD1C
```

Contoh: Ambil pengidentifikasi pembuatan mesin virtual dari Windows

Anda dapat membuat sampel aplikasi untuk mengambil pengidentifikasi pembuatan mesin virtual dari instans Anda yang menjalankan Windows. Untuk informasi selengkapnya, lihat Mendapatkan pengidentifikasi pembuatan mesin virtual di dokumentasi Microsoft.

# Dokumen identitas instans untuk instans Amazon EC2

Setiap instans yang Anda luncurkan memiliki dokumen identitas instans yang memberikan informasi tentang instans tersebut. Anda dapat menggunakan dokumen identitas instans untuk memvalidasi atribut instans.

Dokumen identitas instans dibuat saat instans dihentikan dan dimulai, dimulai ulang, atau diluncurkan. Anda dapat mengakses dokumen identitas instans untuk instance melalui Layanan Metadata Instance (IMDS). Untuk petunjuknya, lihatMengambil dokumen identitas instance.

Dokumen identitas instance menggunakan format JSON plaintext. Ini termasuk informasi berikut.

| Data | Deskripsi |
| --- | --- |
| `accountId` | ID AWS akun yang meluncurkan instance. |
| `architecture` | Arsitektur AMI yang digunakan untuk meluncurkan instans (i386 \| x86_64 \| arm64). |
| `availabilityZone` | Nama Zona Ketersediaan tempat instance berjalan. Misalnya, `us-east-1` . Perlu diingat bahwa nama Zona Ketersediaan mungkin berbeda di seluruh AWS akun. |
| `billingProducts` | Produk penagihan instans. |
| `devpayProductCodes` | Telah usang. |
| `imageId` | ID AMI yang digunakan untuk meluncurkan instans. |
| `instanceId` | ID instans. |
| `instanceType` | Tipe instans dari instans tersebut. |
| `kernelId` | ID kernel yang terkait dengan instans, jika ada. |
| `marketplaceProductCodes` | Kode AWS Marketplace produk AMI digunakan untuk meluncurkan instance. |
| `pendingTime` | Tanggal dan waktu instans diluncurkan. |
| `privateIp` | Alamat IP pribadi dari instance. Untuk IPv4-only dan instance dual-stack, ini berisi alamat IPv4. Misalnya IPv6-only , ini berisi alamat IPv6. |
| `ramdiskId` | ID dari RAM disk yang terkait dengan instans, jika ada. |
| `region` | Wilayah tempat instans berjalan. |
| `version` | Versi format dokumen identitas instans. |

# Mengambil dokumen identitas instans untuk instans EC2

Dokumen identitas instans untuk instans Amazon EC2 menggunakan format JSON plaintext. Untuk deskripsi isi dokumen identitas instans, lihatthe section called “Dokumen identitas instans”.

Dokumen identitas instans disimpan dalam metadata instans untuk instance, dalam kategori data `instance-identity/document` dinamis. Anda mengakses dokumen identitas instans untuk instance dengan menghubungkan ke instance dan mengambilnya dari metadata instans.

Anda dapat mengakses metadata instans menggunakan alamat IPv4 169.254.169.254 atau alamat IPv6. fd00:ec2::254 Ini adalahLink-local alamat, artinya Anda dapat mengaksesnya hanya dari instance. Contoh di halaman ini menggunakan alamat IPv4 dari IMDS:. 169.254.169.254 Untuk mengambil metadata instans untuk instans EC2 melalui IPv6, gunakan. fd00:ec2::254

Untuk memverifikasi keaslian dokumen identitas instance setelah Anda mengambilnya, lihatVerifikasi dokumen identitas instans.

IMDSv2

Linux

Jalankan perintah berikut dari instance Linux Anda untuk mengambil dokumen identitas instance.

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/document
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda untuk mengambil dokumen identitas instance.

```
[string]$token = (Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token-ttl-seconds' = '21600'} `
    -Method PUT -Uri 'http://169.254.169.254/latest/api/token' -UseBasicParsing).Content
```

```
(Invoke-WebRequest -Headers @{'X-aws-ec2-metadata-token' = $token} `
    -Uri 'http://169.254.169.254/latest/dynamic/instance-identity/document' -UseBasicParsing).Content
```

IMDSv1

Linux

Jalankan perintah berikut dari instance Linux Anda untuk mengambil dokumen identitas instance.

```
curl http://169.254.169.254/latest/dynamic/instance-identity/document
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda untuk mengambil dokumen identitas instance.

```
(Invoke-WebRequest http://169.254.169.254/latest/dynamic/instance-identity/
document).Content
```

Berikut ini adalah output contoh.

```
{
    "devpayProductCodes" : null,
    "marketplaceProductCodes" : [ "1abc2defghijklm3nopqrs4tu" ],
    "availabilityZone" : "us-west-2b",
    "privateIp" : "10.158.112.84",
    "version" : "2017-09-30",
    "instanceId" : "i-1234567890abcdef0",
    "billingProducts" : null,
    "instanceType" : "t2.micro",
    "accountId" : "123456789012",
    "imageId" : "ami-5fb8c835",
    "pendingTime" : "2016-11-19T16:32:11Z",
    "architecture" : "x86_64",
    "kernelId" : null,
    "ramdiskId" : null,
    "region" : "us-west-2"
}
```

## Verifikasi dokumen identitas instans untuk instans Amazon EC2

Jika Anda bermaksud menggunakan konten dokumen identitas instans untuk tujuan penting, Anda harus memverifikasi konten dan keaslian sebelum menggunakannya.

Dokumen identitas instans plaintext disertai dengan tiga tanda tangan yang di-hash dan dienkripsi. Anda dapat menggunakan tanda tangan ini untuk memverifikasi asal dan keaslian dokumen identitas instans serta informasi yang disertakan. Tanda tangan berikut disediakan:

- Base64-encoded Tanda tangan—ini adalah hash SHA256 yang dikodekan base64 dari dokumen identitas instance yang dienkripsi menggunakan pasangan kunci RSA.
- Tanda tangan PKCS7—Ini adalah hash SHA1 dari dokumen identitas instans yang dienkripsi menggunakan pasangan kunci DSA.
- RSA-2048 Tanda tangan—ini adalah hash SHA256 dari dokumen identitas instance yang dienkripsi menggunakan pasangan kunci. RSA-2048

Setiap tanda tangan tersedia di titik akhir yang berbeda dalam metadata instans. Anda dapat menggunakan salah satu dari tanda tangan ini, tergantung persyaratan hashing dan enkripsi Anda. Untuk memverifikasi tanda tangan, Anda harus menggunakan sertifikat AWS publik yang sesuai.

Opsi

- Opsi 1: Verifikasi dokumen identitas instance menggunakan tanda tangan PKCS7
- Opsi 2: Verifikasi dokumen identitas instance menggunakan tanda tangan yang dikodekan base64
- Opsi 3: Verifikasi dokumen identitas instance menggunakan RSA-2048 tanda tangan

## Opsi 1: Verifikasi dokumen identitas instance menggunakan tanda tangan PKCS7

Topik ini menjelaskan cara memverifikasi dokumen identitas instance menggunakan tanda tangan PKCS7 dan sertifikat publik AWS DSA.

Instans Linux

Untuk memverifikasi dokumen identitas instance menggunakan tanda tangan PKCS7 dan AWS Sertifikat publik DSA

1. Hubungkan dengan instans.
2. Ambil tanda tangan PKCS7 dari metadata instans dan tambahkan ke file baru bernama `pkcs7` beserta header dan footer yang diperlukan. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

IMDSv2

```
$ echo "-----BEGIN PKCS7-----" >> pkcs7 \
 && TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600"` \
 && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/pkcs7 >> pkcs7 \
 && echo "" >> pkcs7 \
 && echo "-----END PKCS7-----" >> pkcs7
```

IMDSv1

```
$ echo "-----BEGIN PKCS7-----" >> pkcs7 \
 && curl -s http://169.254.169.254/latest/dynamic/instance-identity/pkcs7
 >> pkcs7 \
 && echo "" >> pkcs7 \
 && echo "-----END PKCS7-----" >> pkcs7
```

3. Temukan sertifikat publik DSA untuk Wilayah Anda di AWS sertifikat publik misalnya tanda tangan dokumen identitas dan tambahkan konten ke file baru yang bernama `certificate`.
4. Gunakan perintah OpenSSL smime untuk memverifikasi tanda tangan. Masukkan opsi `-verify` untuk menunjukkan bahwa tanda tangan perlu diverifikasi, dan opsi `-noverify` untuk menunjukkan bahwa sertifikat tidak perlu diverifikasi.

```
$ openssl smime -verify -in pkcs7 -inform PEM -certfile certificate -noverify | tee
 document
```

Jika tanda tangan valid, pesan `Verification successful` muncul.

Perintah tersebut juga menulis konten dokumen identitas instans ke file baru bernama `document`. Anda dapat membandingkan konten dokumen identitas instans dari metadata instans dengan konten file ini menggunakan perintah berikut.

```
$ openssl dgst -sha256 < document
```

```
$ curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/document | openssl dgst -sha256
```

Jika tanda tangan tidak dapat diverifikasi, kontak Dukungan.

Instans Windows

Prasyarat

Prosedur ini membutuhkan kelas `System.Security` Microsoft.NET Core. Untuk menambahkan kelas ke PowerShell sesi Anda, jalankan perintah berikut.

```
PS C:\> Add-Type -AssemblyName System.Security
```

> **Note**
>
> Perintah menambahkan kelas ke PowerShell sesi saat ini saja. Jika Anda memulai sesi baru, Anda harus menjalankan perintah lagi.

Untuk memverifikasi dokumen identitas instance menggunakan tanda tangan PKCS7 dan AWS Sertifikat publik DSA

1. Hubungkan dengan instans.
2. Ambil tanda tangan PKCS7 dari metadata instans, konversi ke byte array, dan tambahkan ke variabel bernama `$Signature`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

   IMDSv2

   ```
   PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-ec2-
   metadata-token-ttl-seconds' = '21600'} -Uri http://169.254.169.254/latest/api/
   token).Content
   ```

   ```
   PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} -Uri http://169.254.169.254/latest/
   dynamic/instance-identity/pkcs7).Content)
   ```

IMDSv1

```
PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Uri
 http://169.254.169.254/latest/dynamic/instance-identity/pkcs7).Content)
```

3. Ambil dokumen identitas instans plaintext dari metadata instans, ubah menjadi array byte, dan tambahkan ke variabel bernama `$Document`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

   IMDSv2

   ```
   PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} -Uri http://169.254.169.254/latest/
   dynamic/instance-identity/document).Content)
   ```

   IMDSv1

   ```
   PS C:\> $Document =  [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
    http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
   ```

4. Temukan sertifikat publik DSA untuk Wilayah Anda di AWS sertifikat publik misalnya tanda tangan dokumen identitas dan tambahkan konten ke file baru yang bernama `certificate.pem`.
5. Ekstrak sertifikat dari file sertifikat dan simpan dalam variabel bernama `$Store`.

   ```
   PS C:\> $Store =
    [Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptog
   Path certificate.pem)))
   ```

6. Verifikasi tanda tangan.

   ```
   PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
   ```

   ```
   PS C:\> $SignatureDocument.Decode($Signature)
   ```

   ```
   PS C:\> $SignatureDocument.CheckSignature($Store, $true)
   ```

Jika tanda tangan valid, perintah tidak mengembalikan keluaran. Jika tanda tangan tidak dapat diverifikasi, perintah menampilkan `Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer`. Jika tanda tangan tidak dapat diverifikasi, hubungi AWS Dukungan.

7. Validasi konten dokumen identitas instans.

```
PS C:
\> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

Jika konten dokumen identitas instans valid, perintah mengembalikan `True`. Jika dokumen identitas instans tidak dapat divalidasi, hubungi AWS Dukungan.

## Opsi 2: Verifikasi dokumen identitas instance menggunakan tanda tangan yang dikodekan base64

Topik ini menjelaskan cara memverifikasi dokumen identitas instance menggunakan tanda tangan yang dikodekan base64 dan sertifikat publik AWS RSA.

Instans Linux

Untuk memvalidasi dokumen identitas instance menggunakan tanda tangan yang dikodekan base64 dan AWS Sertifikat publik RSA

1. Hubungkan dengan instans.
2. Ambil tanda tangan berenkode base64 dari metadata instans, konversikan ke biner, dan tambahkan tanda tangan tersebut ke file bernama `signature`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

   IMDSv2

```
$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600"` \
 && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/signature | base64 -d >> signature
```

IMDSv1

```
$ curl -s http://169.254.169.254/latest/dynamic/instance-identity/signature |
 base64 -d >> signature
```

3. Ambil dokumen identitas instans plaintext dari metadata instans dan tambahkan ke file bernama `document`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

   IMDSv2

   ```
   $ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
   metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
   dynamic/instance-identity/document >> document
   ```

   IMDSv1

   ```
   $ curl -s http://169.254.169.254/latest/dynamic/instance-identity/document
    >> document
   ```

4. Tambahkan sertifikat publik RSA untuk Wilayah Anda di AWS sertifikat publik misalnya tanda tangan dokumen identitas dan tambahkan konten ke file baru yang bernama `certificate`.
5. Ekstrak kunci publik dari sertifikat publik AWS RSA dan simpan ke file bernama`key`.

   ```
   $ openssl x509 -pubkey -noout -in certificate >> key
   ```

6. Gunakan perintah OpenSSL dgst untuk memverifikasi dokumen identitas instans.

   ```
   $ openssl dgst -sha256 -verify key -signature signature document
   ```

   Jika tanda tangannya valid, file`Verification successful` pesan muncul.

   Perintah tersebut juga menulis konten dokumen identitas instans ke file baru bernama `document`. Anda dapat membandingkan konten dokumen identitas instans dari metadata instans dengan konten file ini menggunakan perintah berikut.

   ```
   $ openssl dgst -sha256 < document
   ```

```
$ curl -s -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/document | openssl dgst -sha256
```

Jika tanda tangan tidak dapat diverifikasi, kontak Dukungan.

Instans Windows

Untuk memvalidasi dokumen identitas instance menggunakan tanda tangan yang dikodekan base64 dan AWS Sertifikat publik RSA

1. Hubungkan dengan instans.
2. Ambil tanda tangan berenkode base64 dari metadata instans, konversikan ke array bita, dan tambahkan tanda tangan tersebut ke variabel bernama `$Signature`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

   IMDSv2

   ```
   PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-
   ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/
   token).Content
   ```

   ```
   PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
   instance-identity/signature).Content)
   ```

   IMDSv1

   ```
   PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest
    http://169.254.169.254/latest/dynamic/instance-identity/signature).Content)
   ```

3. Ambil dokumen identitas instans plaintext dari metadata instans, ubah menjadi array byte, dan tambahkan ke variabel bernama `$Document`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

IMDSv2

```
PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
 @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
instance-identity/document).Content)
```

IMDSv1

```
PS C:\> $Document =  [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
 http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. Tambahkan sertifikat publik RSA untuk Wilayah Anda di AWS sertifikat publik misalnya tanda tangan dokumen identitas dan tambahkan konten ke file baru yang bernama `certificate.pem`.
5. Verifikasi dokumen identitas instans.

```
PS C:\> [Security.Cryptography.X509Certificates.X509Certificate2]::new((Resolve-
Path certificate.pem)).PublicKey.Key.VerifyData($Document, 'SHA256', $Signature)
```

Jika tanda tangan valid, perintah mengembalikan `True`. Jika tanda tangan tidak dapat diverifikasi, kontak Dukungan.

## Opsi 3: Verifikasi dokumen identitas instance menggunakan RSA-2048 tanda tangan

Topik ini menjelaskan cara memverifikasi dokumen identitas instance menggunakan RSA-2048 tanda tangan dan sertifikat AWS RSA-2048 publik.

Instans Linux

Untuk memverifikasi dokumen identitas instance menggunakan RSA-2048 tanda tangan dan AWS RSA-2048 sertifikat publik

1. Hubungkan dengan instans.
2. Ambil RSA-2048 tanda tangan dari metadata instance dan tambahkan ke file bernama `rsa2048` sepanjang header dan footer yang diperlukan. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

IMDSv2

```
$ echo "-----BEGIN PKCS7-----" >> rsa2048 \
 && TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600"` \
 && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
dynamic/instance-identity/rsa2048 >> rsa2048 \
 && echo "" >> rsa2048 \
 && echo "-----END PKCS7-----" >> rsa2048
```

IMDSv1

```
$ echo "-----BEGIN PKCS7-----" >> rsa2048 \
 && curl -s http://169.254.169.254/latest/dynamic/instance-identity/rsa2048
 >> rsa2048 \
 && echo "" >> rsa2048 \
 && echo "-----END PKCS7-----" >> rsa2048
```

3. Temukan sertifikat RSA-2048 publik untuk Wilayah Anda di AWS sertifikat publik misalnya tanda tangan dokumen identitas dan tambahkan konten ke file baru bernama`certificate`.
4. Gunakan perintah OpenSSL smime untuk memverifikasi tanda tangan. Masukkan opsi `-verify` untuk menunjukkan bahwa tanda tangan perlu diverifikasi, dan opsi `-noverify` untuk menunjukkan bahwa sertifikat tidak perlu diverifikasi.

   ```
   $ openssl smime -verify -in rsa2048 -inform PEM -certfile certificate -noverify |
    tee document
   ```

   Jika tanda tangan valid, pesan `Verification successful` muncul. Jika tanda tangan tidak dapat diverifikasi, kontak Dukungan.

Instans Windows

Prasyarat

Prosedur ini membutuhkan kelas `System.Security` Microsoft.NET Core. Untuk menambahkan kelas ke PowerShell sesi Anda, jalankan perintah berikut.

```
PS C:\> Add-Type -AssemblyName System.Security
```

> **Note**
>
> Perintah menambahkan kelas ke PowerShell sesi saat ini saja. Jika Anda memulai sesi baru, Anda harus menjalankan perintah lagi.

Untuk memverifikasi dokumen identitas instance menggunakan RSA-2048 tanda tangan dan AWS RSA-2048 sertifikat publik

1. Hubungkan dengan instans.
2. Ambil RSA-2048 tanda tangan dari metadata instance, ubah menjadi array byte, dan tambahkan ke variabel bernama`$Signature`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

   IMDSv2

   ```
   PS C:\> [string]$token = (Invoke-WebRequest -Method Put -Headers @{'X-aws-
   ec2-metadata-token-ttl-seconds' = '21600'} http://169.254.169.254/latest/api/
   token).Content
   ```

   ```
   PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
   instance-identity/rsa2048).Content)
   ```

   IMDSv1

   ```
   PS C:\> $Signature = [Convert]::FromBase64String((Invoke-WebRequest
    http://169.254.169.254/latest/dynamic/instance-identity/rsa2048).Content)
   ```

3. Ambil dokumen identitas instans plaintext dari metadata instans, ubah menjadi array byte, dan tambahkan ke variabel bernama `$Document`. Gunakan salah satu perintah berikut, tergantung pada versi IMDS yang digunakan oleh instans.

   IMDSv2

   ```
   PS C:\> $Document = [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest -Headers
    @{'X-aws-ec2-metadata-token' = $Token} http://169.254.169.254/latest/dynamic/
   instance-identity/document).Content)
   ```

IMDSv1

```
PS C:\> $Document =  [Text.Encoding]::UTF8.GetBytes((Invoke-WebRequest
 http://169.254.169.254/latest/dynamic/instance-identity/document).Content)
```

4. Temukan sertifikat RSA-2048 publik untuk Wilayah Anda di AWS sertifikat publik misalnya tanda tangan dokumen identitas dan tambahkan konten ke file baru bernama`certificate.pem`.
5. Ekstrak sertifikat dari file sertifikat dan simpan dalam variabel bernama `$Store`.

```
PS C:\> $Store =
 [Security.Cryptography.X509Certificates.X509Certificate2Collection]::new([Security.Cryptog
Path certificate.pem)))
```

6. Verifikasi tanda tangan.

```
PS C:\> $SignatureDocument = [Security.Cryptography.Pkcs.SignedCms]::new()
```

```
PS C:\> $SignatureDocument.Decode($Signature)
```

```
PS C:\> $SignatureDocument.CheckSignature($Store, $true)
```

Jika tanda tangan valid, perintah tidak mengembalikan keluaran. Jika tanda tangan tidak dapat diverifikasi, perintah menampilkan `Exception calling "CheckSignature" with "2" argument(s): "Cannot find the original signer`. Jika tanda tangan tidak dapat diverifikasi, hubungi AWS Dukungan.

7. Validasi konten dokumen identitas instans.

```
PS C:
\> [Linq.Enumerable]::SequenceEqual($SignatureDocument.ContentInfo.Content, $Document)
```

Jika konten dokumen identitas instans valid, perintah mengembalikan `True`. Jika dokumen identitas instans tidak dapat divalidasi, hubungi AWS Dukungan.

# AWS sertifikat publik misalnya tanda tangan dokumen identitas

Sertifikat AWS publik berikut dapat digunakan untuk memverifikasi isi dokumen identitas instans seperti yang dijelaskan dalamVerifikasi dokumen identitas instans.

Pastikan bahwa Anda menggunakan sertifikat yang benar untuk Wilayah Anda dan untuk prosedur verifikasi yang Anda gunakan. Jika Anda memverifikasi tanda tangan PKCS7, gunakan sertifikat DSA. Jika Anda memverifikasi tanda tangan yang dikodekan base64, gunakan sertifikat RSA. Jika Anda memverifikasi RSA-2048 tanda tangan, gunakan RSA-2048 sertifikat.

Perluas setiap Wilayah di bawah ini untuk melihat Region-specific sertifikat.

AS Timur (Virginia Utara) — us-timur-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUE1y2NIKCU+Rg4uu4u32koG9QEYIwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MzQwMVoXDTI5MDQyODE3MzQwMVowXDELMAkGA1UEBhMCVVMx
```

```
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUE1y2NIKC
U+Rg4uu4u32koG9QEYIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAlxSmwcWnhT4uAeSinJuz+1BTcKhVSWb5jT8pYjQb8ZoZkXXRGb09mvYeU
NeqOBr27rvRAnaQ/9LUQf72+SahDFuS4CMI8nwowytqbmwquqFr4dxA/SDADyRiF
ea1UoMuNHTY49J/1vPomqsVn7mugTp+TbjqCfOJTpu0temHcFA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALFpzEAVWaQZMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
ODU5MTJaGA8yMTk1MDExNzA4NTkxMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAjS2vqZu9mEOhOq+0bRpAbCUiapbZMFNQqRg7kTlr7Cf+gDqXKpHPjsng
SfNz+JHQd8WPI+pmNs+q0Z2aTe23klmf2U52KH9/j1k8RlIbap/yFibFTSedmegX
E5r447GbJRsHUmuIIfZTZ/oRlpuIIO5/Vz7SOj22tdkdY2ADp7caZkNxhSP915fk
2jJMTBUOzyXUS2rBU/ulNHbTTeePjcEkvzVYPahD30TeQ+/A+uWUu89bHSQOJR8h
Um4cFApzZgN3aD5j2LrSMu2pctkQwf9CaWyVznqrsGYjYOY66LuFzSCXwqSnFBfv
fFBAFsjCgY24G2DoMyYkF3MyZlu+rwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUrynSPp4uqSECwy+PiO4qyJ8TWSkwgY4GA1UdIwSBhjCBg4AUrynS
Pp4uqSECwy+PiO4qyJ8TWSmhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALFpzEAVWaQZMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADW/s8lXijwdP6NkEoH1m9XLrvK4YTqkNfR6
er/uRRgTx2QjFcMNrx+g87gAml11z+D0crAZ5LbEhDMs+JtZYR3ty0HkDk6SJM85
haoJNAFF7EQ/zCp1EJRIkLLsC7bcDL/Eriv1swt78/BB4RnC9W9kSp/sxd5svJMg
N9a6FAplpNRsWAnbP8JBlAP93oJzblX2LQXgykTghMkQO7NaY5hg/H5o4dMPclTK
lYGqlFUCH6A2vdrxmpKDLmTn5//5pujdD2MN0df6sZWtxwZ0osljV4rDjm9Q3VpA
NWIsDEcp3GUB4proOR+C7PNkY+VGODitBOw09qBGosCBstwyEqY=
-----END CERTIFICATE-----
```

### AS Timur (Ohio) — us-east-2

#### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

#### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUVJTc+hOU+8Gk3JlqsX438Dk5c58wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MTE0OVoXDTI5MDQyODE3MTE0OVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUVJTc+hOU
+8Gk3JlqsX438Dk5c58wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAywJQaVNWJqW0R0T0xVOSoN1GLk9x9kKEuN67RN9CLin4dA97qa7Mr5W4P
FZ6vnh5CjOhQBRXV9xJUeYSdqVItNAUFK/fEzDdjf1nUfPlQ3OJ49u6CV01NoJ9m
usvY9kWcV46dqn2bk2MyfTTgvmeqP8fiMRPxxnVRkSzlldP5Fg==
```

```
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAM07oeX4xevdMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjA2MTAx
MjU4MThaGA8yMTk1MTExNDEyNTgxOFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA6v6kGMnRmFDLxBEqXzP4npnL65OO0kmQ7w8YXQygSdmNIoScGSU5wfh9
mZdcvCxCdxgALFsFqPvH8fqiE9ttI0fEfuZvHOs8wUsIdKr0Zz0MjSx3cik4tKET
ch0EKfMnzKOgDBavraCDeX1rUDU0Rg7HFqNAOry3uqDmnqtk00XC9GenS3z/7ebJ
fIBEPAam5oYMVFpX6M6St77WdNE8wEU8SuerQughiMVx9kMB07imeVHBiELbMQ0N
lwSWRL/61fA02keGSTfSp/0m3u+lesf2VwVFhqIJs+JbsEscPxOkIRlzy8mGd/JV
ONb/DQpTedzUKLgXbw7KtO3HTG9iXQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU2CTGYE5fTjx7gQXzdZSGPEWAJY4wgY4GA1UdIwSBhjCBg4AU2CTG
YE5fTjx7gQXzdZSGPEWAJY6hYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAM07oeX4xevdMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANdqkIpVypr2PveqUsAKke1wKCOSuw1UmH9k
xX1/VRoHbrI/UznrXtPQOPMmHA2LKSTedwsJuorUn3cFH6qNs8ixBDrl8pZwfKOY
IBJcTFBbI1xBEFkZoO3wczzo5+8vPQ60RVqAaYb+iCa1HFJpccC3Ovajfa4GRdNb
n6FYnluIcDbmpcQePoVQwX7W3oOYLB1QLN7fE6H1j4TBIsFdO3OuKzmaifQlwLYt
DVxVCNDabpOr6Uozd5ASm4ihPPoEoKo7Ilp0fOT6fZ41U2xWA4+HF/89UoygZSo7
K+cQ90xGxJ+gmlYbLFR5rbJOLfjrgDAb2ogbFy8LzHo2ZtSe60M=
-----END CERTIFICATE-----
```

## AS Barat (California Utara) — us-west-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
```

```
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUK2zmY9PUSTR7rc1k2OwPYu4+g7wwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MDI0M1oXDTI5MDQyODE3MDI0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUK2zmY9PU
STR7rc1k2OwPYu4+g7wwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQA1Ng4QmN4n7iPh5CnadSOc0ZfM7by0dBePwZJyGvOHdaw6P6E/vEk76KsC
Q8p+akuzVzVPkU4kBK/TRqLp19wEWoVwhhTaxHjQ1tTRHqXIVlrkw4JrtFbeNM21
GlkSLonuzmNZdivn9WuQYeGe7nUD4w3q9GgiF3CPorJe+UxtbA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJANNPkIpcyEtIMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
OTAzMDdaGA8yMTk1MDQwMzA5MDMwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEApHQGvHvq3SVCzDrC7575BW7GWLzcj8CLqYcL3YY7JffupZ7OjcftO57Z
4fo5Pj0CaS8DtPzh8+8vdwUSMbiJ6cDd3ooio3MnCq6DwzmsY+pY7CiI3UVG7KcH
```

```
4TriDqr1Iii7nB5MiPJ8wTeAqX89T3SYaf6Vo+4GCb3LCDGvnkZ9TrGcz2CHkJsj
AIGwgopFpwhIjVYm7obmuIxSIUv+oNH0wXgDL029Zd98SnIYQd/njiqkzE+lvXgk
4h4Tu17xZIKBgFcTtWPky+POGu81DYFqiWVEyR2JKKm2/iR1dL1YsT39kbNg47xY
aR129sS4nB5Vw3TRQA2jL0ToTIxzhQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUgepyiONs8j+q67dmcWu+mKKDa+gwgY4GA1UdIwSBhjCBg4AUgepy
iONs8j+q67dmcWu+mKKDa+ihYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJANNPkIpcyEtIMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAGLFWyutf1u0xcAc+kmnMPqtc/Q6b79VIX0E
tNoKMI2KR8lcV8ZElXDb0NC6v8UeLpe1WBKjaWQtEjL1ifKg9hdY9RJj4RXIDSK7
33qCQ8juF4vep2U5TTBd6hfWxt1Izi88xudjixmbpUU4YKr8UPbmixldYR+BEx0u
B1KJi9l1lxvuc/Igy/xeHOAZEjAXzVvHp8Bne33VVwMiMxWECZCiJxE4I7+Y6fqJ
pLLSFFJKbNaFyXlDiJ3kXyePEZSc1xiWeyRB2ZbTi5eu7vMG4i3AYWuFVLthaBgu
lPfHafJpj/JDcqt2vKUKfur5edQ6j1CGdxqqjawhOTEqcN8m7us=
-----END CERTIFICATE-----
```

## AS Barat (Oregon) — us-west-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
```

```
MIIDITCCAoqgAwIBAgIUFx8PxCkbHwpD31bOyCtyz3GclbgwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE3MjM1OVoXDTI5MDQyODE3MjM1OVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUFx8PxCkb
HwpD31bOyCtyz3GclbgwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBzOl+9Xy1+UsbUBI95HO9mbbdnuX+aMJXgG9uFZNjgNEbMcvx+h8P9IMko
z7PzFdheQQ1NLjsHH9mSR1SyC4m9ja6BsejH5nLBWyCdjfdP3muZM4O5+r7vUa1O
dWU+hP/T7DUrPAIVMOE7mpYa+WPWJrN6BlRwQkKQ7twm9kDalA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALZL3lrQCSTMMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
OTAxMzJaGA8yMTk1MDExNzA5MDEzMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA02Y59qtAA0a6uzo7nEQcnJ26OKF+LRPwZfixBH+EbEN/Fx0gYy1jpjCP
s5+VRNg6/WbfqAsV6X2VSjUKN59ZMnMY9ALA/Ipz0n00Huxj38EBZmX/NdNqKm7C
qWu1q5kmIvYjKGiadfboU8wLwLcHo8ywvfgI6FiGGsEO9VMC56E/hL6Cohko11LW
dizyvRcvg/IidazVkJQCN/4zC9PUOVyKdhW33jXy8BTg/QH927QuNk+ZzD7HH//y
tIYxDhR6TIZsSnRjz3bOcEHxt1nsidc65mY0ejQty4hy7ioSiapw316mdbtE+RTN
fcH9FPIFKQNBpiqfAW5Ebp3Lal3/+wIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU7coQx8Qnd75qA9XotSWT3IhvJmowgY4GA1UdIwSBhjCBg4AU7coQ
x8Qnd75qA9XotSWT3IhvJmqhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALZL3lrQCSTMMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFZ1e2MnzRaXCaLwEC1pW/f0oRG8nHrlPZ9W
OYZEWbh+QanRgaikBNDtVTwARQcZm3z+HWSkaIx3cyb6vM0DSkZuiwzm1LJ9rDPc
aBm03SEt5v8mcc7sXWvgFjCnUpzosmky6JheCD4O1Cf8k0olZ93FQnTrbg62OK0h
83mGCDeVKU3hLH97FYoUq+3N/IliWFDhvibAYYKFJydZLhIdlCiiB99AM6Sg53rm
oukS3csyUxZyTU2hQfdjyo1nqW9yhvFAKjnnggiwxNKTTPZzstKW8+cnYwiiTwJN
```

```
QpVoZdt0SfbuNnmwRUMi+QbuccXweav29QeQ3ADqjgB0CZdSRKk=
-----END CERTIFICATE-----
```

### Afrika (Cape Town) — af-south-1

#### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7DCCAqwCCQCncbCtQbjuyzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTA2MDQxMjQ4MDVaFw00
NTA2MDQxMjQ4MDVaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbYwggErBgcqhkjOOAQBMIIBHgKBgQC12Nr1gMrHcFSZ7S/A
pQBSCMHWmn2qeoQTMVWqe50fnTd0zGFxDdIjKxUK58/8zjWG5uR4TXRzmZpGpmXB
bSufAR6BGqud2LnT/HIWGJAsnX2uOtSyNfCoJigqwhea5w+CqZ6I7iBDdnB4TtTw
qO6TlnExHFVj8LMkylZgiaE1CQIVAIhdobse4K0QnbAhCL6R2euQzloXAoGAV/21
WUuMz/79Ga0JvQcz1FNy1sT0pU9rU4TenqLQIt5iccn/7EIfNtvVO5TZKulIKq7J
gXZr0x/KIT8zsNweetLOaGehPIYRMPX0vunMMR7hN7qA7W17WZv/76adywIsnDKq
ekfe15jinaX8MsKUdyDK7Y+ifCG4PVhoM4+W2XwDgYQAAoGAIxOKbVgwLxbn6Pi2
6hBOihFv16jKxAQI0hHzXJLV0Vyv9QwnqjJJRfOCy3dB0zicLXiIxeIdYfvqJr+u
hlN8rGxEZYYJjEUKMGvsc0DW85jonXz0bNfcP0aaKH0lKKVjL+OZi5n2kn9wgdo5
F3CVnMl8BUra8A1Tr2yrrE6TVZ4wCQYHKoZIzjgEAwMvADAsAhQfa7MCJZ+/TEY5
AUr0J4wm8VzjoAIUSYZVu2NdRJ/ERPmDfhW5EsjHlCA=
-----END CERTIFICATE-----
```

#### RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAKumfZiRrNvHMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTExMjcw
NzE0MDVaGA8yMTk5MDUwMjA3MTQwNVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQDFd571nUzVtke3rPyRkYfvs3jh0C0EMzzG72boyUNjnfw1+m0TeFraTLKb9T6F
7TuB/ZEN+vmlYqr2+5Va8U8qLbPF0bRH+FdaKjhgWZdYXxGzQzU3ioy5W5ZM1VyB
7iUsxEAlxsybC3ziPYaHI42UiTkQNahmoroNeqVyHNnBpQIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAAJLylWyElEgOpW4B1XPyRVD4pAds8Guw2+krgqkY0HxLCdjosuH
RytGDGN+q75aAoXzW5a7SGpxLxk6Hfv0xp3RjDHsoeP0i1d8MD3hAC5ezxS4oukK
s5gbPOnokhKTMPXbTdRn5ZifCbWlx+bYN/mTYKvxho7b5SVg2o1La9aK
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAIFI+O5A6/ZIMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTA2MDQx
MjQ4MDRaGA8yMTk4MTEwNzEyNDgwNFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAy7/WHBBHOrk+20aumT07g8rxrSM0UXgki3eYgKauPCG4Xx//vwQbuZwI
oeVmR9nqnhfij2wOcQdbLandh0EGtbxerete3IoXzd1KXJb11PVmzrzyu5SPBPuP
iCeV4qdjjkXo2YWM6t9YQ911hcG96YSp89TBXFYUh3KLxfqAdTVhuC0NRGhXpyii
j/czo9njofHhqhTr7UEyPun8NVS2QWctLQ86N5zWR3Q0GRoVqqMrJs0cowHTrVw2
9Qr7QBjjBOVbyYmtYxm/DtiKprYV/e6bCAVok015X1sZDd3oCOQNoGlv5XbHJe2o
JFD8GRRy2rkWO/lNwVFDcwec6zC3QwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCE
goqzjpCpmMgCpszFHwvRaSMbspKtK7wNImUjrSBOfBJsfFulyg1Zgn2nDCK7kQhx
jMJmNIvXbps3yMqQ2cHUkKcKf5t+WldfeT4Vk1Rz6HSA8sd0kgVcIesIaoy2aaXU
VEB/oQziRGyKdN1d4TGYVZXG44CkrzSDvlbmfiTq5tL+kAieznVF3bzHgPZW6hKP
EXC3G/IXrXicFEe6YyE1Rakl62VncYSXiGe/i2XvsiNH3Qlmnx5XS7W0SCN0oAxW
EH9twibauv82DVg1WOkQu8EwFw8hFde9X0Rkiu0qVcuU8lJgFEvPWMDFU5sGB6ZM
gkEKTzMvlZpPbBhg99Jl
-----END CERTIFICATE-----
```

## Asia Pasifik (Hong Kong) – ap-east-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQCO7MJe5Y3VLjAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTAyMDMwMjIxMjFaFw00
NTAyMDMwMjIxMjFaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbgwggEsBgcqhkjOOAQBMIIBHwKBgQDvQ9RzVvf4MAwGbqfX
blCvCoVb9957OkLGn/04CowHXJ+vTBR7eyIa6AoXltsQXBOmrJswToFKKxT4gbuw
jK7s9QQX4CmTRWcEgO2RXtZSVjOhsUQMh+yf7Ht4OVL97LWnNfGsX2cwjcRWHYgI
7lvnuBNBzLQHdSEwMNq0Bk76PwIVAMan6XIEEPnwr4e6u/RNnWBGKd9FAoGBAOCG
eSNmxpW4QFu4pIlAykm6EnTZKKHT87gdXkAkfoC5fAfOxxhnE2HezZHp9Ap2tMV5
8bWNvoPHvoKCQqwfm+OUBlAxC/3vqoVkKL2mG1KgUH9+hrtpMTkwO3RREnKe7I5O
x9qDimJpOihrL4I0dYvy9xUOoz+DzFAW8+ylWVYpA4GFAAKBgQDbnBAKSxWr9QHY
6Dt+EFdGz6lAZLedeBKpaP53Z1DTO34J0C55YbJTwBTFGqPtOLxnUVDlGiD6GbmC
80f3jvogPR1mSmGsydbNbZnbUEVWrRhe+y5zJ3g9qs/DWmDW0deEFvkhWVnLJkFJ
```

```
9pdOu/ibRPH1lE2nz6pK7GbOQtLyHTAJBgcqhkjOOAQDAzAAMC0CFQCoJlwGtJQC
cLoM4p/jtVFOj26xbgIUUS4pDKyHaG/eaygLTtFpFJqzWHc=
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIICSzCCAbQCCQDtQvkVxRvK9TANBgkqhkiG9w0BAQsFADBqMQswCQYDVQQGEwJV
UzETMBEGA1UECBMKV2FzaGluZ3RvbjEQMA4GA1UEBxMHU2VhdHRsZTEYMBYGA1UE
ChMPQW1hem9uLmNvbSBJbmMuMRowGAYDVQQDExFlYzIuYW1hem9uYXdzLmNvbTAe
Fw0xOTAyMDMwMzAwMDZaFw0yOTAyMDIwMzAwMDZaMGoxCzAJBgNVBAYTAlVTMRMw
EQYDVQQIEwpXYXNoaW5ndG9uMRAwDgYDVQQHEwdTZWF0dGxlMRgwFgYDVQQKEw9B
bWF6b24uY29tIEluYy4xGjAYBgNVBAMTEWVjMi5hbWF6b25hd3MuY29tMIGfMA0G
CSqGSIb3DQEBAQUAA4GNADCBiQKBgQC1kkHXYTfc7gY5Q55JJhjTieHAgacaQkiR
Pity9QPDE3b+NXDh4UdP1xdIw73JcIIG3sG9RhWiXVCHh6KkuCTqJfPUknIKk8vs
M3RXflUpBe8Pf+P92pxqPMCz1Fr2NehS3JhhpkCZVGxxwLC5gaG0Lr4rFORubjYY
Rh84dK98VwIDAQABMA0GCSqGSIb3DQEBCwUAA4GBAA6xV9f0HMqXjPHuGILDyaNN
dKcvplNFwDTydVg32MNubAGnecoEBtUPtxBsLoVYXCOb+b5/ZMDubPF9tU/vSXuo
TpYM5Bq57gJzDRaBOntQbX9bgHiUxw6XZWaTS/6xjRJDT5p3S1E0mPI3lP/eJv4o
Ezk5zb3eIf10/sqt4756
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAMoxixvs3YssMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xODA3MjAw
ODQ0NDRaGA8yMTk3MTIyMzA4NDQ0NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA4T1PNsOg0FDrGlWePoHeOSmOJTA3HCRy5LSbYD33GFU2eBrOIxoU/+SM
rInKu3GghAMfH7WxPW3etIAZiyTDDU5RLcUq2Qwdr/ZpXAWpYocNc/CEmBFtfbxF
z4uwBIN3/drM0RSbe/wP9EcgmNUGQMMZWeAji8sMtwpOblNWAP9BniUG0Flcz6Dp
uPovwDTLdAYT3TyhzlohKL3f6O48TR5yTaV+3Ran2SGRhyJjfh3FRpP4VC+z5LnT
WPQHN74Kdq35UgrUxNhJraMGCzznolUuoR/tFMwR93401GsM9fVA7SW3jjCGF81z
PSzjy+ArKyQqIpLW1YGWDFk3sf08FQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDK
2/+C3nPMgtyOFX/I3Cyk+Pui44IgOwCsIdNGwuJysdqp5VIfnjegEu2zIMWJSKGO
lMZoQXjffkVZZ97J7RNDW06oB7kj3WVE8a7U4WEOfnO/CbMUf/x99CckNDwpjgW+
K8V8SzAsQDvYZs2KaE+18GFfLVF1TGUYK2rPSZMHyX+v/TIlc/qUceBycrIQ/kke
jDFsihUMLqgmOV2hXKUpIsmiWMGrFQV4AeV0iXP8L/ZhcepLf1t5SbsGdUA3AUY1
3If8s81uTheiQjwY5t9nM0SY/1Th/tL3+RaEI79VNEVfG1FQ8mgqCK0ar4m0oZJl
tmmEJM7xeURdpBBx36Di
```

```
-----END CERTIFICATE-----
```

## Asia Pasifik (Hyderabad) — ap-south-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC8DCCArCgAwIBAgIGAXjrQ4+XMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24g
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYUAAoGBAJCKGBBoxIUxqBk94JHhwZZbgvbP0DAOoHENQWxp/98lI7/
YOfYJOVMJS22aCnHDurofmo5rvNIkgXi7Rztbhu
+lko9rK6DgpmpUwBU0WZtf34aZ2IWNBwHaVhHvWAQf9/46u18dMa2YucK1Wi+Vc+M
+KldrvgXmhym6ErNlzhJyMAkGByqGSM44BAMDLwAwLAIUaaPKxaOHoYvwz709xXpsQueIq+UCFFa/
GpzoDOSokl1057NU/2hnsiW4
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDCzCCAnSgAwIBAgIJAPw1mjt5QguhMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0yNjA1Mjgx
MDQzMjlaFw0zMTA1MjcxMDQzMjlaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
08B7tDQ/rBXKG6473AGJtDzUQ/H+EdYwM76C7dvqhnjkwMjtR7bY+nJrGDdwD9YT
eYmLqaRYmLJdnjT2vwg9FYNAQHwxZVea846QNI+0yWm1YeJI+yb7Q/sWDIusBEAO
MekjzjWODnmdvvKPx0vJ7nKVg1XT0TunIllJTIY5g5sCAwEAAaOB1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR0OBBYEFM4QY6Q5jpIb3pUP7EBEDUVlvGgsMIGOBgNVHSME
gYYwgYOAFM4QY6Q5jpIb3pUP7EBEDUVlvGgsoWCkXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEOCCQD8NZo7eUILoTASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAGPVdLHdjH2PVcfoK41bRQCp
vzwwOB0XpQ1AqQBYZXmm6+Ggc3nQLoOKLG1oRlc38psWtxBjGRL+ktvyAjgwk+BL
xM7Yx+wDf8m2nfrvIdaMHk0i+U7bWaiFod3cD6XICfwg/tFz50auWTtlIPcNOXAU
qPuHbYlFgMUlNuNp78nB
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAIVWfPw/X82fMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MDQx
NDMwMjhaGA8yMjAxMTIwODE0MzAyOFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAg29QEFriG+qFEjYW/v62nN70lMJY/Hevx5TtmU/VIYBPQa3HUGTBAbbI
2Tmy8UMpa8kZeaYeI3RAfiQWt0Ws7wUrBuO2Pdp518WDPaJUH7RWEuu1BDDkyZRW
NAMNPCn3ph7Od243IFcLGku7HVekel5poqRpSfojrMasjlf+CvixUeAJbmFoxUHK
kh5unzG2sZyO4wHXcJPQkRf5a8zSTPe9YZP1kXPPEv4p/jTSggaYPxXyS6QVaTlV
zLeLFZ0fesLPMeil3KYQtV7IKLQiEA2F6dxWnxNWQlyMHtdq6PucfEmVx17i/Xza
yNBRo0azY8WUNVkEXrRhp/pU8Nh3GQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU9AO1aZk9RLXk2ZvRVoUxYvQy9uwwgY4GA1UdIwSBhjCBg4AU9AO1
aZk9RLXk2ZvRVoUxYvQy9uyhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAIVWfPw/X82fMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBADexluMRQRftqViahCnauEWGdMvLCBr8A+Yr
6hJq0guoxEk/lahxR137DnfMPuSbi1Rx5QKo7oBrWfG/zsgQUnF2IwHTzwD+i/2m
XCane6FiS5RpK3lGdILq8ZmlhQk+6iI8yoZLr0LCfTh+CLgIKHOknfR5lFzgzAiF
SI8/Q9mm+uvYtSTZECI6Zh57QZPoETAG/yl+9jiOy2lAelqa/k1i+Qo8gMfOc+Pm
dwY7o6fV+oucgRlsdey6VM45LeyILQqv0RXtVzjuowanzmCCFMjgqiO9oZAWu4Oh
+F3unijELo01vZJs8s2N3KGlo3/jtUFTX6RTKShZlAPLwBi5GMI=
-----END CERTIFICATE-----
```

## Asia Pasifik (Jakarta) — ap-southeast-3

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC8DCCArCgAwIBAgIGAXbVDEikMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24g
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYUAAoGBAPjuiEx05N3JQ6cVwntJie67D8OuNo4jGRn
+crEtL7YO0jSVB9zGE1ga
+UgRPIaYETL293S8rTJTVgXAqdpBwfaHC6NUzre8U8iJ8FMNnlP9Gw1oUIlgQBjORyynVJexoB31TDZM
```

```
+/52g9O/bpq1QqNyKbeIgyBBlc1dAtr1QLnsMAkGByqGSM44BAMDLwAwLAIUK8E6RDIRtwK+9qnaTOBhvO/
njuQCFFocyT1OxK+UDR888oNsdgtif2Sf
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUMDbUg+reaqO28n34vFUuk6gFWbgwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI2MDQyMjEyMDY0MFoXDTMxMDQyMTEyMDY0MFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCnCS/Vbt0gQ1ebWcur2hSO7PnJifE4OPxQ7RgSAlc4/spJp1sD
P+ZrS0LO1ZJfKhXf1R9S3AUwLnsc7b+IuVXdY5LK9RKqu64nyXP5dx170zoL8loE
yCSuRR2fs+04i2QsWBVP+KFNAn7P5L1EHRjkgTO8kjNKviwRV+OkP9ab5wIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUsnsEPOUYPQO0Gv9t4arIC2NN
kHQwgZkGA1UdIwSBkTCBjoAUsnsEPOUYPQO0Gv9t4arIC2NNkHShYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUMDbUg+re
aqO28n34vFUuk6gFWbgwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQCFibNMMdZF13osIpWxjb7uXrqpDYFat+196JVSdq+WMYLfcQtudZOcTclo
Zh+OXR49nRv9x/Q4XRfMWGCdptp4q/6NCScEUPBbY2+7CJyH0gyC6vYcGmLfJbUi
kMB27EWJ4tm18a/qZockIeGgJgqt5/KSZb5iN5CuxRHRDGZI+Q==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAMtdyRcH51j9MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA0MDgx
MjM5MTZaGA8yMjAxMDkxMjEyMzkxNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAvUsKCxoH6KXRYJLeYTWAQfaBQeCwhJaR56mfUeFHJE4g8aFjWkiN4uc1
TvOyYNnIZKTHWmzmulmdinWNbwP0GiROHb/i7ro0Hhvnptyycgt8ag8affiIbx5X
7ohdwSN2KJ6G0IKflIx7f2NEI0oAMM/9k+T1eVF+MVWzpZoiDp8frLNkqp8+RAgz
ScZsbRfwv3u/if5xJAvdg2nCkIWDMSHEVPoz0lJo7v0ZuDtWWsL1LHnL5ozvsKEk
+ZJyEi23r+U1hIT1NTBdp4yoigNQexedtwCSr7q36oOdDwvZpqYlkLi3uxZ4ta+a
01pzOSTwMLgQZSbKWQrpMvsIAPrxoQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU1GgnGdNpbnL3lLF30Jomg7Ji9hYwgY4GA1UdIwSBhjCBg4AU1Ggn
GdNpbnL3lLF30Jomg7Ji9hahYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
```

```
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAMtdyRcH51j9MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACVl00qQlatBKVeiWMrhpczsJroxDxlZTOba
6wTMZk7c3akb6XMOSZFbGaifkebPZqTHEhDlrClM2j9AIlYcCx6YCrTf4cuhn2mD
gcJN33143eOWSaeRY3ee4j+V9ne98y3kO2wLz95VrRgclPFR8po2iWGzGhwUi+FG
q8dXeCH3N0DZgQsSgQWwmdNQXZZej6RHLU/8In5trHKLY0ppnLBjn/UZQbeTyW5q
RJB3GaveXjfgFUWj2qOcDuRGaikdS+dYaLsi5z9cA3FolHzWxx9MOs8io8vKqQzV
XUrLTNWwuhZy88cOlqGPxnoRbw7TmifwPw/cunNrsjUUOgs6ZTk=
-----END CERTIFICATE-----
```

## Asia Pasifik (Malaysia) — ap-southeast-5

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQC5X6U+vgoLEDAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDAxMDMxMjU3NTRaGA8y
MDUwMDEwMzEyNTc1NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0
b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIg
U2VydmljZXMgTExDMIIBtjCCASsGByqGSM44BAEwggEeAoGBAIzEMPCIPFfOYCg4
BCjKGyl60wOfmwHPzSOXZ3Z2wS/LYNHUtHGwtVNePSTyCu/CuZF6gC9n/wBORtQp
+SSkn+weGc/BmUA1mp/vrN7v+aSCgKJoO+sgpa1PPOgNvUaMw6O5odsZWQCMSjkU
6RTo/PL2v/tMfiCocF4ghvyRC6hvAhUA0VoObKC2IXzXgVvRRupo4qHbcm8CgYAe
bbNuawh3rAxkFvUs9FPzW5E+xllG16Z//61PENKqonmk+zBiBdiIlS1F6ZqmTqkI
z5+qfSt1m3pb3j2W0NT7lEDFvy8Gr6Y2vohCHmL+T1u1Yy4PeqbgrFwcen7y7WoO
/KCV7Y9/ODQMMyuAzT3h5wJNweT7L5MUN8JYpZSi3QOBhAACgYBqaDuG2u6V91Qj
K2wEAElxaaRaNo/ewg/wWDmHYqoeHOR0HfuFCYgASE9f7ULqYtX1VURcgcjw9XN4
BDmPilXvfi04INPTnw4IxFJKDzzC0kVH7esVas982Po8v3megH32H9Rl87r7UG1c
ZEbkSkKVX6YKYg1PR3rfjXgdwVZv/zAJBgcqhkjOOAQDAzAAMC0CFFWeRe2fYW2i
6mMd26Wzbx87Y0DXAhUAoPCnF+5hGJw0jT9aL7QsgcfLi9Y=
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAMuBl6rhZCJKMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDAxMDMx
MjU3NTRaGA8yMjAzMDYwOTEyNTc1NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
```

```
gQDFuKydxZsordNH7bLwIluEGOkX7/CdLdpeqkDKEhQkFwzpRxaX4EAlkGh2/o7D
8qneC9cGQhqSG5WVVBrmZG7sfkFOM4m1AtY++kfv+MYto1VFgLk1xJbkpq1r4YeQ
Ul+ZsJYsZpyX/t+g8s7rW0OVcBsYx4L75bf34z38mwK8PQIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBADD9C4pWL8RUvF1CJW8kExj35xmozlFlmrKs8Zpi8+Eg6q+W9dgd
xMdH95tgZtmVMDqlvVR+DK0iO1BNpqPjrqWkk2tTLivpS+sGzCE/jCl18Q28Rk71
/A3gLD7Rtbq5TKNvuFCHwYMjrTDHI6aBjIaAlDm4e2/j/OxVtHyZGTre
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANC3xtbPhQ2GMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDAxMDMx
MjU3NTRaGA8yMjAzMDYwOTEyNTc1NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAt3aMy7Hsp4ySG3mlfi+pdTcZw6H6XNU1Y36fNdi4c+MzinQQbnqMPyt7
QLgU+XCWmcWsVo7GQF6n9N01Rh+UXXUZU4jcXlFocQPCWf9O+IIIPXkd67kFMUV
HAxCELjfxHbC+I8e7dwOJhmdF4Bfi52Ty8zz0HdE8JDypPkTDlXuGvTgDyW7NP56
I/vlQaXLoYSbcQe5pv2a9gyBaaCM1QoeqwAHAeCNXb9Nuj9ZX3GHGJb3TuqAeKCD
5i9TscCB9XjY6Fx+zfSAobjBZwgLEtLOwJhbZnKmx4gJMaaNFipAjVT2FSS3+yev
eTYBoa1dvhk0ivQyQIPpHmihrmkWuwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBC
fdgyI8GjmCqiHAlh+L1bj0LdNql9zl7RXmOEzsuRdtdMumkxYXX88UtROy3fdi1i
VaEwHdAK8ThzRkesgHza/cXzqCMewaYxujSI6p6G7x99FFeGif1x0FJdj8AoeTL7
4h9bmS/6l4/NL7DJI9G7ovES/hoUA9v9TDhv+vauxXlgfrp0MPecprxBYlrc+DH2
adGCdKCp2lQ2YDKOD9TCEjYIli8XSoyevoWHUjfDYrCrCp8l4s/p7HOgYr8fJBAs
EuVy8211LVz1/X4EMBRNtNjXK9sklsxAOX14NDfBFSS0toxl3K6Tf9t/PviBl95d
hncyDAcFgDCK4w8LLlVW
-----END CERTIFICATE-----
```

## Asia Pasifik (Melbourne) — ap-southeast-4

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjWF7P2MAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
```

```
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAPRXSsQP9E3dw8QXKlrgBgEVCprLHdK/bbrMas0XMu1EhOD
+q
+0PcTr8+iwbtoXlY5MCeatWIp1GrXQjVqsF8vQqxlEuRuYKbR3nq4mWwaeGlx9AG5EjQHRa3GQ44wWHOdof0M3NRI1MP
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUUOsUa3Aj7pphfmDjsZAFNWODCUowDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI2MDYwMzEwNDM1M1oXDTMxMDYwMjEwNDM1M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQDHezwQr2VQpQSTW5TXNefiQrP+qWTGAbGsPeMX4hBMjAJUKys2
NIRcRZaLM/BCew2FIPVjNtlaj6Gwn9ipU4Mlz3zIwAMWi1AvGMSreppt+wV6MRtf
Ojh0Dvj/veJe88aEZJMozNgkJFRS+WFWsckQeL56tf6kY6QTlNo8V/0CsQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUVCy+PqHsYEzvAHcGK+lrlknS
UvowgZkGA1UdIwSBkTCBjoAUVCy+PqHsYEzvAHcGK+lrlknSUvqhYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUUOsUa3Aj
7pphfmDjsZAFNWODCUowEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBRI72kM+Q6PCU584L6UZ8rk6uERCeq5g8dDzlLZMaMpV4n/s2MJbFP2hO+
ovahJnxLTrs7jP3VXZeVj5Eb0dfYneg0+zNtycnHe+9jH+GTKS8+QkwGj3zmwxqG
mZ9V6XoGwtO/aEz24Yp7CRf8etCYEnVI2ChD7xClRV3DJwliEA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAN4GTQ64zVs8MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTMx
MzMzMDBaGA8yMjAxMTIxNzEzMzMwMFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA2BYgeCr+Rk/jIAEDOHS7wJql62vc83QEwjuzk0qOFEReIZzlN1fBRNXK
g0T178Kd3gLYcE59wEFbTe/X5yOA1Lo95x1anSAo7R+Cisf9C2HQuJp+gVb+zx71
lniPF7gHziGpm0M8DdAU/IW+wkZwGbP4z7Hq9+bJ0P2ltvPJ5yxSgkFuDsI9VBHa
CLoprHsCHh2VdP8KcMgQQMmHe1NmBpyTk0ul/aLmQkCQEX6ZIRGOeq228fwlh/t+
Ho+jv87duihVKic6MrL32SlD+maX0LSDUydWdaOLLTGkh7oV7+bFuH6msrXUu+Ur
```

```
ZEP1r/MidCWMhfgrFzeTBzOHA97qxQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUcHMd1cHqzmsQ5hpUK3EMLhHdsi4wgY4GA1UdIwSBhjCBg4AUcHMd
1cHqzmsQ5hpUK3EMLhHdsi6hYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAN4GTQ64zVs8MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAI4PFyVN+7EGSObioiPnv0LL0f70SSzUZJ8p
XO90d4rWea7jIbgZ2AKb+ErynkU9xVg7XQQ5k6KDWgp/4jYFL2dqnt/YAY4PS0un
RSrYElawxLT0BcLn4rcSDC79vQe1xGC5//wDdV6b399COAHRAK6axWYy5w32u9PL
uw0cIp3Ch8JoNwcgTHKRRGzePmBeR4PNqhHTArG4/dJk6/aUO4OpX0WzI6L67CGY
6Nex3dau+gkLCK93dTEkrXtyXHu4wB0J9zd1w+iQ0SEa9eKc78/NjEsF/FZdGrWC
t57lIMOOXJhQ1kRgSwNeZdQWV1dRakvO6sfcvVYkfj1wAvZvvAw=
-----END CERTIFICATE-----
```

## Asia Pasifik (Mumbai) - ap-south-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUDLA+x6tTAP3LRTr0z6nOxfsozdMwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
```

```
MB4XDTI0MDQyOTE0MTMwMVoXDTI5MDQyODE0MTMwMVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUDLA+x6tT
AP3LRTr0z6nOxfsozdMwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAZ7rYKoAwwiiH1M5GJbrT/BEk3OO2VrEPw8ZxgpqQ/EKlzMlOs/0Cyrmp7
UYyUgYFQe5nq37Z94rOUSeMgv/WRxaMwrLlLqD78cuF9DSkXaZIX/kECtVaUnjk8
BZx0QhoIHOpQocJUSlm/dLeMuE0+0A3HNR6JVktGsUdv9ulmKw==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAPRYyD8TtmC0MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjAzMDcx
MDQ1MDFaGA8yMTk1MDgxMTEwNDUwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA0LSS5I/eCT2PM0+qusorBx67QL26BIWQHd/yF6ARtHBb/1DdFLRqE5Dj
07Xw7eENC+T79mOxOAbeWg91KaODOzw6i9I/2/HpK0+NDEdD6sPKDA1d45jRra+v
CqAjI+nV9Vw91wv7HjMk3RcjWGziM8/hw+3YNIutt7aQzZRwIWlBpcqx3/AFd8Eu
2UsRMSHgkGUW6UzUF+h/U8218XfrauKNGmNKDYUhtmyBrHT+k6J0hQ4pN7fe6h+Z
w9RVHm24BGhlLxLHLmsOIxvbrF277uX9Dxu1HfKfu5D2kimTY7xSZDNLR2dt+kNY
/+iWdIeEFpPT0PLSILt52wP6stF+3QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBI
E6w+WWC2gCfoJO6c9HMyGLMFEpqZmz1n5IcQt1h9iyO7Vkm1wkJiZsMhXpk73zXf
TPxuXEacTX3SOEa07OIMCFwkusO5f6leOyFTynHCzBgZ3U0UkRVZA3WcpbNB6Dwy
h7ysVlqyT9WZd7EOYm5j5oue2G2xdei+6etgn5UjyWm6liZGrcOF6WPTdmzqa6WG
ApEqanpkQd/HM+hUYex/ZS6zEhd4CCDLgYkIjlrFbFb3pJ1OVLztIfSN5J4Oolpu
JVCfIq5u1NkpzL7ys/Ub8eYipbzI6P+yxXiUSuF0v9b98ymczMYjrSQXIf1e8In3
OP2CclCHoZ8XDQcvvKAh
-----END CERTIFICATE-----
```

## Asia Pasifik (Selandia Baru) — ap-southeast-6

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq4CCQCcP+D4eLFOajAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDEyMDMxMjEzMDZaGA8y
MDUwMTIwNDEyMTMwNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0
b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIg
U2VydmljZXMgTExDMIIBtjCCASsGByqGSM44BAEwggEeAoGBANBMmM/LbAX13IeB
3yr2pMMpiTW1b1go/axNLfcxEmj0fThKk8y0FAT1frN/UPdI9qRInsyBdEZQLiIH
4WHCXk8IidQLh4DED3G9DvSFFC4Gb8ax+UIQY11wfHPWO4+wA9q8liYTg/FxdJMM
tZS9tZAZxjH6+8yfV8Bt0wUd2Y+lAhUAwg0CxCmZPinHtKbFnpMQfKnzjCMCgYAf
i+D7nGVFOWJ8EYOpRYbG0pHkrpwNy+h01csn6IAwQbIfeOG+SF4N3LeuOAC9vXGY
VGh9XNxuasvqkRAzgRmJ/EYM5Xa/YN62EhddRuN8wCOg66MeczUDh3RTb+Tu6HK0
UR6oTSOQ6rxA5aA1qTc7TqI0+pRZv+QYIZK/ozsZzgOBhAACgYAbVm+rOX8J38cG
LxKjUXVM0KB69SksGLEqjO4ps80EcZ/0W3g1hGLnSX/RjYfAS60hbKK1GX9PpcaO
4mDzCTm3dZ6shbae5t8bBU+mUi8BmaK+Pdu3z5J77y4zv22/yXte8ulZ5J0zKP9K
QP3p3pa7XI9NaXpKGT3yZsJ0BWY2xTAJBgcqhkjOOAQDAzAAMC0CFBs4KQmrxHjQ
OHh0r8JqB8UiwlutAhUAwgxid7wT6FNm8Zwq9hioQ2v2K0A=
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAMwLuv2/tk9eMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDEyMDMx
MjEzMDZaGA8yMjA0MDUwOTEyMTMwNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQC9AsBNOUfT8QpguHqMa9KIXP/YgZOmfRAgEmnLawXTC+40bXUq/fwkx+cebv3x
CvUx6d07/iDlD0dm2Hf2HhPFi3OaCI7vc8c3lqVBvcn6yzmwv35aSEvtVm2NHC3j
4/q6vN2tR2d4a1Mf1QhmOVOJ7XiTEbCy2Yv+pmDbk29V6QIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAHatyQIy612kio8neI4r5VBg8pgKwW+mFGpAzOQ2NOyLCNoOeD1D
lPRaSipBN7WW6jpIozbz7JG7Kqc3N+piPGGzfjH9gZjqUqfOGrNHB5nsxvMHI2iL
e+DwsOcV7qnEVRgYbmLYOVKJv25wCsM/Tcn/6f6scEX2x+H0hAh7rwiP
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
```

```
MIIDOzCCAiOgAwIBAgIJAI/xUfWwSQOtMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDEyMDMx
MjEzMDZaGA8yMjA0MDUwOTEyMTMwNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEArbid9Tnunu0pWCQUqZCkGeiF8hWixi+LBS1AOew2UsT9keAt2rXXGlU/
wSMriKHsvcrfKtuVKj5chArsOyFvagszH5lIKhI+GfWVQEd2aZiOvAJ+iRUQq1+R
gOr2gyjWlf3znZXTcyhHcJoXDhbn5e+kEa3pwS4yKNRpM3P61V4M2702xis08hjP
qMDxwXMa1Fiy4b7nAyNQifgrwb3bCA5k5prmkl/iajaWa1zMMFgLI9vA6UwUSbTK
H9iOuL/xNfMX+Xzq4CCaQYiJ8zdi56kqnB/lrA05NSjOPLskg6sUGkyiDxGK4gzC
DFrO6oygcdlBaxnyAECGdJcOM6TOmwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBg
8yVPNpctrvSvTazf/Ij8qC6+q6df1SptdYKQkOYWeUVZR7+3uTZ9Yiyq+Bh8Hbyf
NcR0eR6UqhgiB3p5slYNhTvLkpuMuqTpVJFvQtzZycnmKXAQ35G5zCYV0+pkF59L
ch26SukWk0+VP3kIM9VmaIZQKJEBlBLcnbDZyXlHUDnZAYPcxxKQ8uFPkW0GOFMr
ekFCGOUmFSItFXOHy5hD2xmctKNk3l12m6cKydYvONtk/3DovKNnw3c5k40WHQW/
Zdh/LWd4/Jrkan+kQnqm4rkrXF+x3aF0qrKTtHMICvpkZ8g7/588hBs4mZnCmFLa
q/g9WqIcd9SMmBkT6KZ8
-----END CERTIFICATE-----
```

## Asia Pasifik (Osaka) — ap-northeast-3

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUHTRhxHhBZFOGvTFKxHoy9+f5Hl8wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2NTQwN1oXDTI5MDQyODE2NTQwN1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUHTRhxHhB
ZFOGvTFKxHoy9+f5Hl8wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAUZx7DcYbhWNTD4BNGhr5beruT2OUoGHH9J73UKxwdqeb9bH1LIWhIZO0X
/1mjn3bWBgCwfoS8gjZwsVB6fZbNBRy8urdBZJ87xF/4JPBjt7S9oGx/zthDUYrC
yK0Y0v4G0PgiS81CvYLg09LpmYhLSJbXENlkC04v5yxdKxZxyg==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAMn1yPk22ditMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNzA3MTkx
MTEyNThaGA8yMTk2MTIyMjExMTI1OFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEArznEYef8IjhrJoazI0QGZkmlmHm/4rEbyQbMNifxjsDE8YWtHNwaM91z
zmyK6Sk/tKlWxcnl3g31iq305ziyFPEewe5Qbwf1iz2cMsvfNBcTh/E6u+mBPH3J
gvGanqUJt6c4IbipdEouIjjnynyVWd4D6erLl/ENijeR1OxVpaqSW5SBK7jms49E
pw3wtbchEl3qsE42Ip4IYmWxqjgaxB7vps91n4kfyzAjUmklcqTfMfPCkzmJCRgp
Vh1C79vRQhmriVKD6BXwfZ8tG3a7mijeDn7kTsQzgO07Z2SAE63PIO48JK8HcObH
tXORUQ/XF1jzi/SIaUJZT7kq3kWl8wIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBj
ThtO9dLvU2QmKuXAhxXjsIdlQgGG3ZGh/Vke4If1ymgLx95v2Vj9Moxk+gJuUSRL
BzFte3TT6b3jPolbECgmAorjj8NxjC17N8QAAI1d0S0gI8kqkG7V8iRyPIFekv+M
pcai1+cIv5IV5qAz8QOMGYfGdYkcoBjsgiyvMJu/2N2UbZJNGWvcEGkdjGJUYYOO
NaspCAFm+6HA/K7BD9zXB1IKsprLgqhiIUgEaW3UFEbThJT+z8UfHG9fQjzzfN/J
nT6vuY/0RRu1xAZPyh2gr5okN/s6rnmh2zmBHU1n8cbCc64MVfXe2g3EZ9Glq/9n
izPrI09hMypJDP04ugQc
```

```
-----END CERTIFICATE-----
```

### Asia Pasifik (Seoul) — ap-northeast-2

#### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

#### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUbBSn2UIO6vYk4iNWV0RPxJJtHlgwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTEzMzg0NloXDTI5MDQyODEzMzg0NlowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUbBSn2UIO
6vYk4iNWV0RPxJJtHlgwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
```

```
AAOBgQAmjTjalG8MGLqWTC2uYqEM8nzI3px1eo0ArvFRsyqQ3fgmWcQpxExqUqRy
l3+2134Kv8dFab04Gut5wlfRtc2OwPKKicmv/IXGN+9bKFnQFjTqif08NIzrDZch
aFT/uvxrIiM+oN2YsHq66GUhO2+xVRXDXVxM/VObFgPERbJpyA==
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANuCgCcHtOJhMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA5MTQx
NTU3NDRaGA8yMTk1MDIxNzE1NTc0NFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA66iNv6pJPmGM20W8HbVYJSlKcAg2vUGx8xeAbzZIQdpGfkabVcUHGB6m
Gy59VXDMDlrJckDDk6dxUOhmcX9z785TtVZURq1fua9QosdbTzX4kAgHGdp4xQEs
mO6QZqg5qKjBP6xr3+PshfQ1rB8Bmwg0gXEm22CC7o77+7N7Mu2sWzWbiUR7vil4
9FjWS8XmMNwFTlShp4l1TDTevDWW/uYmC30RThM9S4QPvTZ0rAS18hHVam8BCTxa
LHaVCH/Yy52rsz0hM/FlghnSnK105ZKj+b+KIp3adBL8OMCjgc/Pxi0+j3HQLdYE
32+FaXWU84D2iP2gDT28evnstzuYTQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQC1
mA4q+12pxy7By6g3nBk1s34PmWikNRJBwOqhF8ucGRv8aiNhRRye9lokcXomwo8r
KHbbqvtK85l0xUZp/Cx4sm4aTgcMvfJP29jGLclDzeqADIvkWEJ4+xncxSYVlS9x
+78TvF/+8h9U2LnSl64PXaKdxHy2IsHIVRN4GtoaP2Xhpa1S0M328Jykq/571nfN
1WRD1c/fQf1edgzRjhQ4whcAhv7WRRF+qTbfQJ/vDxy8lkiOsvU9XzUaZ0fZSfXX
wXxZamQbONvFcxVHY/0PSiM8nQoUmkkBQuKleDwRWvkoJKYKyr3jvXK7HIWtMrO4
jmXe0aMy3thyK6g5sJVg
-----END CERTIFICATE-----
```

## Asia Pasifik (Singapura) — ap-southeast-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
```

```
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUSqP6ih+++5KF07NXngrWf26mhSUwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE0MzAxNFoXDTI5MDQyODE0MzAxNFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUSqP6ih++
+5KF07NXngrWf26mhSUwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQAw13BxW11U/JL58j//Fmk7qqtrZTqXmaz1qm2WlIpJpW750MOcP4ux1uPy
eM0RdVZ4jHSMv5gtLAv/PjExBfw9n6vNCk+5GZG4Xec5DoapBZHXmfMo93sjxBFP
4x9rWn0GuwAVO9ukjYPevq2RerilrqSVvppHtbATVNY2qecXDA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAJVMGw5SHkcvMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
ODU3MTlaGA8yMTk1MDQwMzA4NTcxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAlaSSLfBl7OgmikjLReHuNhVuvM20dCsVzptUyRbut+KmIEEc24wd/xVy
2RMIrydGedkW4tUjkUyOyfET5OAyT43jTzDPHZTkRSVkYjBdcYbe9o/0Q4P7IVS3
XlvwrUu0qo9nSID0mxMnOoF1l8KAqnn10tQ0W+lNSTkasW7QVzcb+3okPEVhPAOq
```

```
MnlY3vkMQGI8zX4iOKbEcSVIzf6wuIffXMGHVC/JjwihJ2USQ8fq6oy686g54P4w
ROg415kLYcodjqThmGJPNUpAZ7MOc5Z4pymFuCHgNAZNvjhZDA842Ojecqm62zcm
Tzh/pNMNeGCRYq2EQX0aQtYOIj7bOQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU6SSB+3qALorPMVNjToM1Bj3oJMswgY4GA1UdIwSBhjCBg4AU6SSB
+3qALorPMVNjToM1Bj3oJMuhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAJVMGw5SHkcvMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAF/0dWqkIEZKg5rca8o0P0VS+tolJJE/FRZO
atHOeaQbWzyac6NEwjYeeV2kY63skJ+QPuYbSuIBLM8p/uTRIvYM4LZYImLGUvoO
IdtJ8mAzq8CZ3ipdMs1hRqF5GRp8lg4w2QpX+PfhnW47iIOBiqSAUkIr3Y3BDaDn
EjeXF6qS4iPIvBaQQ0cvdddNh/pE33/ceghbkZNTYkrwMyBkQlRTTVKXFN7pCRUV
+L9FuQ9y8mP0BYZa5e1sdkwebydU+eqVzsil98ntkhpjvRkaJ5+Drs8TjGaJWlRw
5WuOr8unKj7YxdL1bv7//RtVYVVi296ldoRUYv4SCvJF11z0OdQ=
-----END CERTIFICATE-----
```

## Asia Pasifik (Sydney) — ap-southeast-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFxWyAdk4oiXIOC9PxcgjYYh71mwwDQYJKoZIhvcNAQEL
```

```
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE1MjE0M1oXDTI5MDQyODE1MjE0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUFxWyAdk4
oiXIOC9PxcgjYYh71mwwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQByjeQe6lr7fiIhoGdjBXYzDfkX0lGGvMIhRh57G1bbceQfaYdZd7Ptc0jl
bpycKGaTvhUdkpMOiV2Hi9dOOYawkdhyJDstmDNKu6P9+b6Kak8He5z3NU1tUR2Y
uTwcz7Ye8Nldx//ws3raErfTI7D6s9m63OX8cAJ/f8bNgikwpw==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAL2bOgb+dq9rMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
OTAwNTdaGA8yMTk1MDQwMzA5MDA1N1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAmRcyLWraysQS8yDC1b5Abs3TUaJabjqWu7d5gHik5Icd6dKl8EYpQSeS
vz6pLhkgO4xBbCRGlgE8LS/OijcZ5HwdrxBiKbicR1YvIPaIyEQQvF5sX6UWkGYw
Ma5IRGj4YbRmJkBybw+AAV9Icb5LJNOMWPi34OWM+2tMh+8L234v/JA6ogpdPuDr
sM6YFHMZ0NWo58MQ0FnEj2D7H58Ti//vFPl0TaaPWaAIRF85zBiJtKcFJ6vPidqK
f2/SDuAvZmyHC8ZBHg1moX9bR5FsU3QazfbW+c+JzAQWHj2AaQrGSCITxCMlS9sJ
l51DeoZBjnx8cnRe+HCaC4YoRBiqIQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU/wHIo+r5U31VIsPoWoRVsNXGxowwgY4GA1UdIwSBhjCBg4AU/wHI
o+r5U31VIsPoWoRVsNXGxoyhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAL2bOgb+dq9rMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACobLvj8IxlQyORTz/9q7/VJL509/p4HAeve
92riHp6+Moi0/dSEYPeFTgdWB9W3YCNc34Ss9TJq2D7t/zLGGlbI4wYXU6VJjL0S
hCjWeIyBXUZOZKFCb0DSJeUElsTRSXSFuVrZ9EAwjLvHni3BaC9Ve34iP71ifr75
8Tpk6PEj0+JwiijFH8E4GhcV5chB0/iooU6ioQqJrMwFYnwo1cVZJD5v6D0mu9bS
TMIJLJKv4QQQqPsNdjiB7G9bfkB6trP8fUVYLHLsVlIy5lGx+tgwFEYkG1N8IOO/
2LCawwaWm8FYAFd3IZl04RImNs/IMG7VmH1bf4swHOBHgCN1uYo=
```

```
-----END CERTIFICATE-----
```

### Asia Pasifik (Taipei) — ap-east-2

#### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7jCCAq4CCQDrb+HArMxssTAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDA1MjQxNDU2MDZaGA8y
MDUwMDUyNTE0NTYwNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0
b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIg
U2VydmljZXMgTExDMIIBtjCCASsGByqGSM44BAEwggEeAoGBAPFb5R5TBUs2N0YW
aJcOjxr4UCDrpVT3x6EeqoGFNAc1yiZw9iv++ff6wsXm0VG5XV1QkWMQcjPfK/2I
TvXh9XEOhsG7vaBHoWkb2gmIxE6C2u8TfoZu0ZmiMceFK7aYz5w/q2J9CmRs38dT
I9qgTq0LYX+VkiNYOQnn7sQXQ8CjAhUAp2Lue9SfYTmubbpH+3j91Sivc+cCgYAi
DiyGU5R5VgiNV3mhuAyiHCft3LEmTIxos/hx/rHzgm+vCMH55tYejIrv9n7YGrIZ
MejOt8bGYxqSf0Ir/aoj3ZfHihxUzwyfyzwH7D8kAodXJNsPUTKi0O0pGl/e+nVD
JhTIFV3AwC8Tt0r2hljbjhOiyhdYy5hxWHa/WokLAQOBhAACgYAkGGHXbFr/JXol
itpS1qfiETWSRkXGHrdSmUkRumgti2y7c57TgXPwCyNlP1sTr1duy/kYDpwSwyqi
qEYfko0b/xWpDEQQPsWBgolpEYP7RIT85IxrTCrC9ohz85n7v6m9oF6BiBQGtZdk
GBN7iqemg85UaF6mXfHvl3tUYjsB2jAJBgcqhkjOOAQDAy8AMCwCFED9Ybn6fEVF
wLNjDvULmki5bPBWAhR64pxsHFRHz0ZQi+SYMWdIpEs5RQ==
-----END CERTIFICATE-----
```

#### RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAJp99iy2CTXhMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDA1MjQx
NDU2MDZaGA8yMjAzMTAyOTE0NTYwNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQDVdclgpcIIsyGG51ag8srK1UMfZ+Ioh9E4PLP+cIWQcdh31fv2kV/whKaqfmif
NefBIuHrTPl6L+mJQbsfeA+TiEx1Jamg4lwQfhgQKQCmjZVMaHtK+AY57TVN6FBf
HXxT2MXzktsKxAQRGeSTFQYrHRzSYvu+GkA+37lqyWRfcwIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBABM+WL9MSb7z3bNOTMkM7ufPucvj+LQ+zEAJiuG0rcB8GI2DhGB5
8ohDPUGlq1cP7vYmDkWG/FrLqx7qgIzkbE3zfb62JpdOuX73oHgCmikMfq8ifoS1
CUxMQ36PDZBFNPzGZxv5kEORqhmGBkYCoIUCQ43jh3aALR/A/rfzfij8
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAJt5hgMuWbpMMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDA1MjQx
NDU2MDZaGA8yMjAzMTAyOTE0NTYwNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA91FXs9Wk9kMI3lkNhjC69C6RTBpSvdLWSohPDlJKXFoaGYIDtyW1ViwO
FORAKz0koarVBx1D4fNKCxf+mhYAf//4Pxt31GzjehHNs2YVGimwhq/SNr+3AuIO
PibA+qScXamaiknE1Pxb4jisbBu2PMAG9OxU2dESNwiqBMvdJvg30YNTUITZ6r0e
MPjR1Y/EFVx/SLHW6wdvwqhNi9hz8HHdThzjCbCQPvmoLBJIKXIfosKqvauSezt1
KJSoS2jMx1OjJB2bBlQTZC8vF0WFNPoiH4jXyJ6+wMUMNSSd5b3uL5stVTMzuIz4
39Hd5YW41rmAy+6jmpGp2eXn6Td0rQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQAs
gcgv+Xr67Cj08jKdAUT4GuCoGxveOFAjSDjs5y0cD+kXIT/Yj/+HKFOx5cwThT8F
TxNQ/oVuo3bGeuFEX799H0rT9Jupy0WiE1Vx81mk2M/1EDtns437edM4ofiWR8tR
Q1VNiRgz6Crr0nFgNoQa9HmR6/M+bImxevHJkWB3skvnxtA+lpKGwJs8jo6fA7AW
jr00opK0r731iBd1z/5hjIjkHn5NizAdiJEh3lVY/A3WS82aygW8LvYnKZeaBfZJ
J4NgxeflxocLFKBJD+hyf5u1hhvl7Mn/aGr5VI8/0F2oUqYo+xsHLT3keNghywO4
NNcyLcD9u16HVXhRhGvo
-----END CERTIFICATE-----
```

## Asia Pasifik (Thailand) — ap-southeast-7

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC8TCCAq8CCQCOEEMWiJIJpTAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDA0MTIxNTI4NTZaGA8y
MDUwMDQxMzE1Mjg1NlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0
b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIg
U2VydmljZXMgTExDMIIBtzCCASsGByqGSM44BAEwggEeAoGBAJWKfrIlz9+/U4wm
NDJPX00avUTdksBKX5RwHkY4o0///2G2HWwiBbxw8F1U+RzCXr80gSoe4+/SCSg3
uDuhJyzN5r4jr7c590CAgsiGdWERs7l4gLQCP504feU4TEjJoq9A8MfRZeqMj7Ug
fT5InTUW7S1/r98ddG/rBHMsJwEbAhUAneIckZ1YeyGcCeL32lujMhj+g68CgYA+
RsyTXCy3Tug5aHun51IfcG2d+pn5K/tv/N3WUR18RplVctrLhwIOUoAsDWPWtxNV
sODetezAyo759CK43JAmYGzXKbRFUHm3n46jP88tSdhuSeJhc/D4l5/0+2L7ndXp
L3+W4N05NdiKSGI8e8t452wTZv/RODivPZ3RtuCDywOBhQACgYEAjVS1Huwzdn1J
+kpd2Rcbe1BAGkmvk5sUu0KhyttqIB1kxTxeWZgsdO8REZSC5gJOwkcGFvXnb3DY
+Ms45r0e0s0rx2FFYjqMLwyRpK9wUjJfSXeJMa9iLQEXuyzBz6zPgfemXbS3zNq/
```

```
eoJ9ztIwjB9DMoKL+E1vSLsTGhehqRowCQYHKoZIzjgEAwMxADAuAhUAmljDM3c9
hf0j4XbMjjpnzrx1xhkCFQCASR9pgFNgLK8y6Kojj+P1KJkrSQ==
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAIuIHAhL0xWcMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDA0MTIx
NTI4NTZaGA8yMjAzMDkxNzE1Mjg1NlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQCajgAeOauwvqGDLrHvxujnZ1BnkMzwjrycMUTkj8jqNtWoDQWUJVNPZJILosEU
VwK2I3oNkEsx/ryl9XfXcNNceoYfVEPzkTzozrZyuOG66FWtUU1LKeJ7h9/rX0Zd
9lZEokrdr6dLPt9FsHWaK5ExlUnWBjNltcQLkkKqoeYaFwIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAE4G5G+FvKTsX3T7BEcId7f5LSCc2J9gZRDiWn2oTr40CrBMOzJT
KsWr9W89YXW3gaGWltzc0WCwYQbJZgAkuEAZItJjbhdnns87ZbsFO+NZhc6gDtjA
WC3dPlSB9b6rfVoVW9O6Xwa7iNXZo8ddYVJ/ZOIv/totUz9qJt4DmmKk
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANaQIrYcijxaMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDA0MTIx
NTI4NTZaGA8yMjAzMDkxNzE1Mjg1NlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAxtMGK4r6WerVtMfpRRCY3DMP9Q/s4jnRIqolPaxeT99BAFp03HXyOrTz
WjggEHArRGGeDowFkL7Vj+QNY7ran2lnYVlAQ7Ow6OLl6WttT6lMsgsLUIcsED6G
eJ1Ko+ovT1qvmuush1U5R2pHkcCJ1IT2s6uQff85l066K7dCFpaoA1QHUk9zACHj
i0DtWuTAWBXirAZSGlmtP6uj1Aw6y5kACDTjIvZJeW/kmswfApTKaJ56eNkJsUBU
NpKwt3um1BMZduy0cjUv0Sc3dIbLNyfOdyPn8tX5u5ck0EPdtBB5WjEh71IXdZJD
oaBREV7AHg5ERPQsm0BqSvMQt09KiwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDG
qJyZbv/GUQnm8dEyVYgilQr3Yu35n8sfTwwZ1OJzIVce7NBPWePMfYii/tvGoKNp
KwFa/Vugbn1EwqpTEKqkPE1dOZvnUa97XxhYkoW5U7sgdKCANi1KWjywn7MiJ3Eg
j4gdqYK8wxvHi21ppqr572UO77uA8YMBoBT/CyQzWYUSmbqwKnnzaQBAZeO2iAk
VHuNU+9UsntNB676gRl9ag3Wfxq3yx5Ee1CeQf+US3HJn/pKk1H8dExXmBHvHwO6
GKUVNPN1rxfjTiaSt8wuO80uElAnyHIM3VOR8rJO7PKsobyEeJV0WI1hURO+wxpL
h3IsW7iBrFVvhX5xx7ZU
-----END CERTIFICATE-----
```

## Asia Pasifik (Tokyo) — ap-northeast-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIULgwDh7TiDrPPBJwscqDwiBHkEFQwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTEyMjMxMFoXDTI5MDQyODEyMjMxMFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIEExMQ4IULgwDh7Ti
DrPPBJwscqDwiBHkEFQwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBtjAglBde1t4F9EHCZOj4qnY6Gigy07Ou54i+lR77MhbpzE8V28Li9l+YT
QMIn6SzJqU3/fIycIro1OVY1lHmaKYgPGSEZxBenSBHfzwDLRmC9oRp4QMe0BjOC
gepj1lUoiN7OA6PtA+ycNlsP0oJvdBjhvayLiuM3tUfLTrgHbw==
```

```
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAL9KIB7Fgvg/MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
OTAwMjVaGA8yMTk1MDExNzA5MDAyNVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAz0djWUcmRW85C5CiCKPFiTIvj6y2OuopFxNE5d3Wtab10bm06vnXVKXu
tz3AndG+Dg0zIL0gMlU+QmrSR0PH2PfV9iejfLak9iwdm1WbwRrCEAj5VxPe0Q+I
KeznOtxzqQ5Wo5NLE9bA61sziUAFNVsTFUzphEwRohcekYyd3bBC4v/RuAjCXHVx
40z6AIksnAOGN2VABMlTeMNvPItKOCIeRLlllSqXX1gbtL1gxSW40JWdF3WPB68E
e+/1U3F7OEr7XqmNODOL6yh92QqZ8fHjG+afOL9Y2Hc4g+P1nk4w4iohQOPABqzb
MPjK7B2Rze0f9OEc51GBQu13kxkWWQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU5DS5IFdU/QwYbikgtWvkU3fDwRgwgY4GA1UdIwSBhjCBg4AU5DS5
IFdU/QwYbikgtWvkU3fDwRihYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAL9KIB7Fgvg/MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAG/N7ua8IE9IMyno0n5T57erBvLTOQ79fIJN
Mf+mKRM7qRRsdg/eumFft0rLOKo54pJ+Kim2cngCWNhkzctRHBV567AJNt4+ZDG5
hDgV0IxWO1+eaLE4qzqWP/9VrO+p3reuumgFZLVpvVpwXBBeBFUf2drUR14aWfI2
L/6VGINXYs7uP8v/2VBS7r6XZRnPBUy/R4hv5efYXnjwA9gq8+a3stC2ur8m5ySl
faKSwE4H320yAyaZWH4gpwUdbUlYgPHtm/ohRtiWPrN7KEG5Wq/REzMIjZCnxOfS
6KR6PNjlhxBsImQhmBvz6j5PLQxOxBZIpDoiK278e/1Wqm9LrBc=
-----END CERTIFICATE-----
```

## Kanada (Pusat) — ca-central-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
```

```
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUIrLgixJJB5C4G8z6pZ5rB0JU2aQwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE1MzU0M1oXDTI5MDQyODE1MzU0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUIrLgixJJ
B5C4G8z6pZ5rB0JU2aQwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBHiQJmzyFAaSYs8SpiRijIDZW2RIo7qBKb/pI3rqK6yOWDlPuMr6yNI81D
IrKGGftg4Z+2KETYU4x76HSf0s//vfH3QA57qFaAwddhKYy4BhteFQl/Wex3xTlX
LiwI07kwJvJy3mS6UfQ4HcvZy219tY+0iyOWrz/jVxwq7TOkCw==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAJNKhJhaJOuMMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjA3Mjkx
MTM3MTdaGA8yMTk2MDEwMjExMzcxN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAhDUh6j1ACSt057nSxAcwMaGr8Ez87VA2RW2HyY8l9XoHndnxmP50Cqld
+26AJtltlqHpI1YdtnZ6OrVgVhXcVtbvte0lZ3ldEzC3PMvmISBhHs6A3SWHA9ln
```

```
InHbToLX/SWqBHLOX78HkPRaG2k0COHpRy+fG9gvz8HCiQaXCbWNFDHZev9OToNI
xhXBVzIa3AgUnGMalCYZuh5AfVRCEeALG60kxMMC8IoAN7+HG+pMdqAhJxGUcMO0
LBvmTGGeWhi04MUZWfOkwn9JjQZuyLg6B1OD4Y6s0LB2P1MovmSJKGY4JcF8Qu3z
xxUbl7Bh9pvzFR5gJN1pjM2n3gJEPwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQAJ
UNKM+gIIHNk0G0tzv6vZBT+o/vt+tIp8lEoZwaPQh1121iw/I7ZvhMLAigx7eyvf
IxUt9/nf8pxWaeGzi98RbSmbap+uxYRynqe1p5rifTamOsguuPrhVpl12OgRWLcT
rjg/K60UMXRsmg2w/cxV45pUBcyVb5h6Op5uEVAVq+CVns13ExiQL6kk3guG4+Yq
LvP1p4DZfeC33a2Rfre2IHLsJH5D4SdWcYqBsfTpf3FQThH0l0KoacGrXtsedsxs
9aRd7OzuSEJ+mBxmzxSjSwM84Ooh78DjkdpQgv967p3d+8NiSLt3/n7MgnUy6WwB
KtDujDnB+ttEHwRRngX7
-----END CERTIFICATE-----
```

### Kanada Barat (Calgary) — ca-barat-1

#### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAYPouptUMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAMITzTJUa6cBsIfdHN69zW/
aHjUB4r1ZfKb1FMhIp9EZtEf5n+06oXjUG2+dKRSlFQeEK333ehNZsPd6uqey6TYKtHpFb5XRLS8BpqB
+7gnbAd0CBZM5o4NWesSQ1GLnTdQcGZkYG/
QESkbadoCXQTifCujJE682hTDLIVt1d4ewwCQYHKoZIzjgEAwMvADAsAhRJc4gRS/HWTkCR2MESaQEe/
jOMNQIUNoTwLvuPrmGPupPlGiHe0veZiO8=
-----END CERTIFICATE-----
```

#### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUCVFIhsz9lHA88TjOEChaY+KhHjkwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI2MDYxNzA5MjE0M1oXDTMxMDYxNjA5MjE0M1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQDK1kIcG5Q6adBXQM75GldfTSiXl7tn54p10TnspI0ErDdb2B6q
2Ji/v4XBVH13ZCMgqlRHMqV8AWI5iO6gFn2A9sN3AZXTMqwtZeiDdebq3k6Wt7ie
```

```
YvpXTg0qvgsjQIovRZWaBDBJy9x8C2hW+w9lMQjFHkJ7Jy/PHCJ69EzebQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUwCJ9341XAe3MNiXb/yszt4om
ccIwgZkGA1UdIwSBkTCBjoAUwCJ9341XAe3MNiXb/yszt4omccKhYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUCVFIhsz9
lHA88TjOEChaY+KhHjkwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQCuorFtp2WcEreUMNlL44zF8JDSkju0ik27J4OdKRow6tPvqYDmyGTvybWh
N4GtOuNEhWzmj4E7eMHzRslxaHXNhxIa8l+M/GRaq9jJwAb6aU9YWcVH9FT/6Jr0
q6myzukFBNLZ9zCrrclp8dOFdGf0310RDpmGmF3qzU0MlHh+Iw==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALyTn5IHrIZjMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMzEyMDcx
NTM3MDFaGA8yMjAzMDUxMzE1MzcwMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA1GP5os424BJmGPCKOSgOclP7lzUiB85du03M4hfjzSOszsBpmBGFDLz1
owYHtIxlq3+Vi1Lt5Qlx3id/ov1QyaBPFWXVeklHVXy9vieCcI3TdjGjTl1W/8MM
m3X26QPcsnHM/Kk2wJ7s186MrqmdSsp3SCPpxv4vEG2Q9yR2bXY41hpc2rWlW8qU
D0JGXluvmmAdFnto20l1XWZ6xFen1h60DRugek/ufCbN+ljKy0xLqPoavH0Ybjsb
UpsAsBs7phaoN+X/5hIERfbp5LfVnqq54pNG5KNu4KynfW9+kA/WS4cJ6FTTN5t+
y0P1HvcL+BL2RuDy6T2bB21xw5WqtQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQURTvu/Dd4zDnmS5G5CfVLnmUBNOswgY4GA1UdIwSBhjCBg4AURTvu
/Dd4zDnmS5G5CfVLnmUBNOuhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALyTn5IHrIZjMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFt523A3Aug6/F8xxyITgA8gkU0btFhlXNSP
U4U2OQ9n0tWI9WqnKNWH3KBxwY5EPitU6b3LM4xc9lDWpz7h2Pto+WhxP9LVKe6f
r8r7teTLCVZ7cfYZHzHg+flZjVpAgzE5BVfrRlj3QKpv0hYT3J1wMtI++Vorq5Nf
aPjzedehJLhmZVALwnfqfLrgv6/gmraP9Vmoa8U4D6AljNiQGYaLwyoPoRm3bUs2
v1Mh9GkEQ1b9+1pFXcqgzJJTGRuiPCyPbECI79FAnx5JM/CkGJV8HlOmjIW1qkK1
Y2qT7wzErrKLJyB53Pw15BdIM1onbDAQreZb0yZQLdoEl/tx7Uk=
-----END CERTIFICATE-----
```

Tiongkok (Beijing) – cn-north-1

DSA

```
-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFADBcMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEMwIBcNMTUwNTEzMDk1OTE1
WhgPMjE5NDEwMTYwOTU5MTVaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNo
aW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24g
V2ViIFNlcnZpY2VzIExMQzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/l
gsw9+QSnEJeYWnmivJWOBdn9CyDpN7cpHVmeGgNJL2fvImWyWe2f2Kq/BL9l7N7C
P2ZT52/sH9orlck1n2zO8xPi7MItgPHQwu3OxsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwHO/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
vtBj/SM4/IgQ3xJslFcl90TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWFaOdy0+OOaECAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAdSzN2+0E
V1BfR3DPWJHWRf1b7zl+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GEtqhZUqteY7
zAceoLrVu/7OynRyfQetJVGichaaxLNM3lcr6kcxOowb+WQQ84cwrB3keykH4gRX
KHB2rlWSxta+2panSEO1JX2q5jhcFP90rDOtZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZlnIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+61lMVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFHbQp1peGC19idOUqxPxWsasWxQXO0azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----
```

RSA

```
-----BEGIN CERTIFICATE-----
MIIDCzCCAnSgAwIBAgIJALSOMbOoU2svMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
uhhUNlqAZdcWWB/OSDVDGk3OA99EFzOn/mJlmciQ/Xwu2dFJWmSCqEAE6gjufCjQ
q3voxAhC2CF+elKtJW/C0Sz/LYo60PUqd6iXF4h+upB9HkOOGuWHXsHBTsvgkgGA
1CGgel4U0Cdq+23eANr8N8m28UzljjSnTlrYCHtzN4sCAwEAAaOB1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR0OBBYEFBkZu3wT27NnYgrfH+xJz4HJaNJoMIGOBgNVHSME
gYYwgYOAFBkZu3wT27NnYgrfH+xJz4HJaNJooWCkXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEOCCQC0jjGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqmzll7e8Hgb
oADS0ph+YUz5P/bUCm61wFjlxaTfwKcuTR3ytj7bFLoW5Bm7Sa+TCl3lOGb2taon
```

```
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUeK+MRiWu+Oh5/lJGii3qw4YByx
SUDlRyNy1jJFstEZjOhs
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAOtrM5XLDSjCMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQx
MDAxNDJaGA8yMTk1MDExNzEwMDE0MlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAvVBz+WQNdPiM9S+aUULOQEriTmNDUrjLWLr7SfaOJScBzis5D5ju0jh1
+qJdkbuGKtFX5OTWTm8pWhInX+hIOoS3exC4BaANoa1A3o6quoG+Rsv72qQf8LLH
sgEi6+LMlCN9TwnRKOToEabmDKorss4zFl7VSsbQJwcBSfOcIwbdRRaW9Ab6uJHu
79L+mBR3Ea+G7vSDrVIA8goAPkae6jY9WGw9KxsOrcvNdQoEkqRVtHo4bs9fMRHU
Etphj2gh4ObXlFN92VtvzD6QBs3CcoFWgyWGvzg+dNG5VCbsiiuRdmii3kcijZ3H
Nv1wCcZoEAqH72etVhsuvNRC/xAP8wIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA8
ezx5LRjzUU9EYWYhyYIEShFlP1qDHs7F4L46/5lc4pL8FPoQm5CZuAF31DJhYi/b
fcV7i3n++/ymQbCLC6kAg8DUB7NrcROll5ag8d/JXGzcTCnlDXLXx1905fPNa+jI
0q5quTmdmiSi0taeaKZmyUdhrB+a7ohWdSdlokEIOtbH1P+g5yll3bI2leYE6Tm8
LKbyfK/532xJPqO9abx4Ddn89ZEC6vvWVNDgTsxERg992Wi+/xoSw3XxkgAryIv1
zQ4dQ6irFmXwCWJqc6kHg/M5W+z60S/94+wGTXmp+19U6Rkq5jVMLh16XJXrXwHe
4KcgIS/aQGVgjM6wivVA
-----END CERTIFICATE-----
```

## Tiongkok (Ningxia) – cn-northwest-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIDNjCCAh4CCQD3yZ1w1AVkTzANBgkqhkiG9w0BAQsFADBcMQswCQYDVQQGEwJV
UzEZMBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEg
MB4GA1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEMwIBcNMTUwNTEzMDk1OTE1
WhgPMjE5NDEwMTYwOTU5MTVaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNo
aW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24g
V2ViIFNlcnZpY2VzIExMQzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
AMWk9vyppSmDU3AxZ2Cy2bvKeK3F1UqNpMuyeriizi+NTsZ8tQqtNloaQcqhto/l
gsw9+QSnEJeYWnmivJWOBdn9CyDpN7cpHVmeGgNJL2fvImWyWe2f2Kq/BL9l7N7C
P2ZT52/sH9orlck1n2zO8xPi7MItgPHQwu3OxsGQsAdWucdxjHGtdchulpo1uJ31
jsTAPKZ3p1/sxPXBBAgBMatPHhRBqhwHO/Twm4J3GmTLWN7oVDds4W3bPKQfnw3r
```

```
vtBj/SM4/IgQ3xJslFcl90TZbQbgxIi88R/gWTbs7GsyT2PzstU30yLdJhKfdZKz
/aIzraHvoDTWFaOdy0+OOaECAwEAATANBgkqhkiG9w0BAQsFAAOCAQEAdSzN2+0E
V1BfR3DPWJHWRf1b7zl+1X/ZseW2hYE5r6YxrLv+1VPf/L5I6kB7GEtqhZUqteY7
zAceoLrVu/7OynRyfQetJVGichaaxLNM3lcr6kcxOowb+WQQ84cwrB3keykH4gRX
KHB2rlWSxta+2panSEO1JX2q5jhcFP90rDOtZjlpYv57N/Z9iQ+dvQPJnChdq3BK
5pZlnIDnVVxqRike7BFy8tKyPj7HzoPEF5mh9Kfnn1YoSVu+61lMVv/qRjnyKfS9
c96nE98sYFj0ZVBzXw8Sq4Gh8FiVmFHbQp1peGC19idOUqxPxWsasWxQXO0azYsP
9RyWLHKxH1dMuA==
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDCzCCAnSgAwIBAgIJALSOMbOoU2svMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0yMzA3MDQw
ODM1MzlaFw0yODA3MDIwODM1MzlaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
uhhUNlqAZdcWWB/OSDVDGk3OA99EFzOn/mJlmciQ/Xwu2dFJWmSCqEAE6gjufCjQ
q3voxAhC2CF+elKtJW/C0Sz/LYo60PUqd6iXF4h+upB9HkOOGuWHXsHBTsvgkgGA
1CGgel4U0Cdq+23eANr8N8m28UzljjSnTlrYCHtzN4sCAwEAAaOB1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR0OBBYEFBkZu3wT27NnYgrfH+xJz4HJaNJoMIGOBgNVHSME
gYYwgYOAFBkZu3wT27NnYgrfH+xJz4HJaNJooWCkXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEOCCQC0jjGzqFNrLzASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBAECji43p+oPkYqmzll7e8Hgb
oADS0ph+YUz5P/bUCm61wFjlxaTfwKcuTR3ytj7bFLoW5Bm7Sa+TCl3lOGb2taon
2h+9NirRK6JYk87LMNvbS40HGPFumJL2NzEsGUeK+MRiWu+Oh5/lJGii3qw4YByx
SUDlRyNy1jJFstEZjOhs
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAPu4ssY3BlzcMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEyMDMy
MTI5MzJaGA8yMTk1MDUwODIxMjkzMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAsOiGi4A6+YTLzCdIyP8b8SCT2M/6PGKwzKJ5XbSBoL3gsnSWiFYqPg9c
uJPNbiy9wSA9vlyfWMd90qvTfiNrT6vewP813QdJ3EENZOx4ERcf/Wd22tV72kxD
```

```
yw1Q3I1OMH4bOItGQAxU5OtXCjBZEEUZooOkU8RoUQOU2Pql4NTiUpzWacNutAn5
HHS7MDc4lUlsJqbN+5QW6fFrcNG/0Mrib3JbwdFUNhrQ5j+Yq5h78HarnUivnX/3
Ap+oPbentv1qd7wvPJu556LZuhfqI0TohiIT1Ah+yUdN5osoaMxTHKKtf/CsSJ1F
w3qXqFJQA0VWsqjFyHXFI32I/GOupwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCn
Um00QHvUsJSN6KATbghowLynHn3wZSQsuS8E0COpcFJFxP2SV0NYkERbXu0n/Vhi
yq5F8v4/bRA2/xpedLWmvFs7QWlomuXhSnYFkd33Z5gnXPb9vRkLwiMSw4uXls35
qQraczUJ9EXDhrv7VmngIk9H3YsxYrlDGEqh/oz4Ze4ULOgnfkauanHikk+BUEsg
/jsTD+7e+niEzJPihHdsvKFDlud5pakEzyxovHwNJ1GS2I//yxrJFIL91mehjqEk
RLPdNse7N6UvSnuXcOokwu6l6kfzigGkJBxkcq4gre3szZFdCQcUioj7Z4xtuTL8
YMqfiDtN5cbD8R8ojw9Y
-----END CERTIFICATE-----
```

### Eropa (Frankfurt) — eu-central-1

#### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

#### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUFD5GsmkxRuecttwsCG763m3u63UwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE1NTUyOVoXDTI5MDQyODE1NTUyOVowXDELMAkGA1UEBhMCVVMx
```

```
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUFD5Gsmkx
RuecttwsCG763m3u63UwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBBh0WaXlBsW56Hqk588MmJxsOrvcKfDjF57RgEDgnGnQaJcStCVWDO9UYO
JX2tdsPw+E7AjDqjsuxYaotLn3Mr3mK0sNOXq9BljBnWD4pARg89KZnZI8FN35HQ
O/LYOVHCknuPL123VmVRNs51qQA9hkPjvw21UzpDLxaUxt9Z/w==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAKD+v6LeR/WrMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
OTA4MTlaGA8yMTk1MDExNzA5MDgxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAka8FLhxs1cSJGK+Q+q/vTf8zVnDAPZ3U6oqppOW/cupCtpwMAQcky8DY
Yb62GF7+C6usniaq/9W6xPn/3o//wti0cNt6MLsiUeHqNl5H/4U/Q/fR+GA8pJ+L
npqZDG2tFi1WMvvGhGgIbScrjR4VO3TuKy+rZXMYvMRk1RXZ9gPhk6evFnviwHsE
jV5AEjxLz3duD+u/SjPp1vloxe2KuWnyC+EKInnka909sl4ZAUh+qIYfZK85DAjm
GJP4W036E9wTJQF2hZJrzsiB1MGyC1WI9veRISd30izZZL6VVXLXUtHwVHnVASrS
zZDVpzj+3yD5hRXsvFigGhY0FCVFnwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUxC2l6pvJaRflgu3MUdN6zTuP6YcwgY4GA1UdIwSBhjCBg4AUxC2l
6pvJaRflgu3MUdN6zTuP6YehYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAKD+v6LeR/WrMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAIK+DtbUPppJXFqQMv1f2Gky5/82ZwgbbfXa
HBeGSii55b3tsyC3ZW5ZlMJ7Dtnr3vUkiWbV1EUaZGOUlndUFtXUMABCb/coDndw
CAr53XTv7UwGVNe/AFO/6pQDdPxXn3xBhF0mTKPrOGdvYmjZUtQMSVb9lbMWCFfs
w+SwDLnm5NF4yZchIcTs2fdpoyZpOHDXy0xgxO1gWhKTnYbaZOxkJvEvcckxVAwJ
obF8NyJla0/pWdjhlHafEXEN8lyxyTTyOa0BGTuYOBD2cTYYynauVKY4fqHUkr3v
Z6fboaHEd4RFamShM8uvSu6eEFD+qRmvqlcodbpsSOhuGNLzhOQ=
-----END CERTIFICATE-----
```

## Eropa (Irlandia) — eu-west-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUakDaQ1Zqy87Hy9ESXA1pFC116HkwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MTgxMFoXDTI5MDQyODE2MTgxMFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUakDaQ1Zq
y87Hy9ESXA1pFC116HkwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQADIkn/MqaLGPuK5+prZZ5Ox4bBZLPtreO2C7r0pqU2kPM2lVPyYYydkvP0
lgSmmsErGu/oL9JNztDe2oCA+kNy17ehcsf8cw0uP861czNFKCeU8b7FgBbL+sIm
qi33rAq6owWGi/5uEcfCR+JP7W+oSYVir5r/yDmWzx+BVH5S/g==
```

```
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAOrmqHuaUt0vMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTEwMjkw
OTA2MTlaGA8yMTk1MDQwMzA5MDYxOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAjE7nVu+aHLtzp9FYV25Qs1mvJ1JXD7J0iQ1Gs/RirW9a5ZECCtc4ssnf
zQHq2JRVr0GRchvDrbm1HaP/avtFQR/Thvfltwu9AROVT22dUOTvERdkNzveoFCy
hf52Rqf0DMrLXG8ZmQPPXPDFAv+sVMWCDftcChxRYZ6mP9O+TpgYNT1krD5PdvJU
7HcXrkNHDYqbsg8A+Mu2hzl0QkvUET83Csg1ibeK54HP9w+FsD6F5W+6ZSHGJ88l
FI+qYKs7xsjJQYgXWfEt6bbckWs1kZIaIOyMzYdPF6ClYzEec/UhIe/uJyUUNfpT
VIsI5OltBbcPF4c7Y20jOIwwI2SgOQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUF2DgPUZivKQR/Zl8mB/MxIkjZDUwgY4GA1UdIwSBhjCBg4AUF2Dg
PUZivKQR/Zl8mB/MxIkjZDWhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAOrmqHuaUt0vMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAGm6+57W5brzJ3+T8/XsIdLTuiBSe5ALgSqI
qnO5usUKAeQsa+kZIJPyEri5i8LEodh46DAF1RlXTMYgXXxl0YggX88XPmPtok17
l4hib/D9/lu4IaFIyLzYNSzsETYWKWoGVe7ZFz60MTRTwY2u8YgJ5dec7gQgPSGj
avB0vTIgoW41G58sfw5b+wjXCsh0nROon79RcQFFhGnvup0MZ+JbljyhZUYFzCli
31jPZiKzqWa87xh2DbAyvj2KZrZtTe2LQ48Z4G8wWytJzxEeZdREe4NoETf+Mu5G
4CqoaPR05KWkdNUdGNwXewydb3+agdCgfTs+uAjeXKNdSpbhMYg=
-----END CERTIFICATE-----
```

## Eropa (London) — eu-west-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
```

```
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUCgCV/DPxYNND/swDgEKGiC5I+EwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MjkxNFoXDTI5MDQyODE2MjkxNFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUCgCV/DPx
YNND/swDgEKGiC5I+EwwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQATPu/sOE2esNa4+XPEGKlEJSgqzyBSQLQc+VWo6FAJhGG9fp7D97jhHeLC
5vwfmtTAfnGBxadfAOT3ASkxnOZhXtnRna460LtnNHm7ArCVgXKJo7uBn6ViXtFh
uEEw4y6p9YaLQna+VC8Xtgw6WKq2JXuKzuhuNKSFaGGw9vRcHg==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANBx0E2bOCEPMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNjA4MTEx
NDU2NDJaGA8yMTk2MDExNTE0NTY0MlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEArYS3mJLGaMrh2DmiPLbqr4Z+xWXTzBWCjOwpsuHE9H6dWUUyl2Bgnu+Z
d8QvW306Yleec45M4F2RA3J4hWHtShzsMlOJVRt+YulGeTf9OCPr26QmIFfs5nD4
```

```
fgsJQEry2MBSGA9Fxq3Cw6qkWcrOPsCR+bHOU0XykdKl0MnIbpBf0kTfciAupQEA
dEHnM2J1L2iI0NTLBgKxy5PXLH9weX20BFauNmHH9/J07OpwL20SN5f8TxcM9+pj
Lbk8h1V4KdIwVQpdWkbDL9BCGlYjyadQJxSxz1J343NzrnDM0M4h4HtVaKOS7bQo
Bqt2ruopLRCYgcuFHck/1348iAmbRQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBG
wujwU1Otpi3iBgmhjMClgZyMMn0aQIxMigoFNqXMUNx1Mq/e/Tx+SNaOEAu0n2FF
aiYjvY0/hXOx75ewzZvM7/zJWIdLdsgewpUqOBH4DXFhbSk2TxggSPb0WRqTBxq5
Ed7F7+7GRIeBbRzdLqmISDnfqey8ufW0ks51XcQNomDIRG5s9XZ5KHviDCar8FgL
HngBCdFI04CMagM+pwTO9XN1Ivt+NzUj208ca3oP1IwEAd5KhIhPLcihBQA5/Lpi
h1s3170z1JQ1HZbDrH1pgp+8hSI0DwwDVb3IIH8kPR/J0Qn+hvOl2HOpaUg2Ly0E
pt1RCZe+W7/dF4zsbqwK
-----END CERTIFICATE-----
```

## Eropa (Milan) — eu-south-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAqwCCQCMElHPdwG37jAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTA0MjkyMDM1MjJaFw00
NTA0MjkyMDM1MjJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbYwggErBgcqhkjOOAQBMIIBHgKBgQDAkoL4YfdMI/MrQ0oL
NPfeEk94eiCQA5xNOnU7+2eVQtEqjFbDADFENh1p3sh9Q9OoheLFH8qpSfNDWn/0
ktCS909ApTY6Esx1ExjGSeQq/U+SC2JSuuTT4WFMKJ63a/czMtFkEPPnVIjJJJmT
HJSKSsVUgpdDIRvJXuyB0zdB+wIVALQ3OLaVGdlPMNfS1nD/Yyn+32wnAoGAPBQ3
7XHg5NLOS4326eFRUT+4ornQFjJjP6dp3pOBEzpImNmZTtkCNNUKE4Go9hv5T4lh
R0pODvWv0CBupMAZVBP9ObplXPCyEIZtuDqVa7ukPOUpQNgQhLLAqkigTyXVOSmt
ECBj9tu5WNP/x3iTZTHJ+g0rhIqpgh012UwJpKADgYQAAoGAV1OEQPYQUg5/M3xf
6vE7jKTxxyFWEyjKfJK7PZCzOIGrE/swgACy4PYQW+AwcUweSlK/Hx2OaZVUKzWo
wDUbeu65DcRdw2rSwCbBTU342sitFo/iGCV/Gjf+BaiAJtxniZze7J1ob8vOBeLv
uaMQmgOYeZ5e0fl04GtqPl+lhcQwCQYHKoZIzjgEAwMwADAtAhQdoeWLrkm0K49+
AeBK+j6m2h9SKQIVAIBNhS2a8cQVABDCQXVXrc0tOmO8
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAOZ3GEIaDcugMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTEwMjQx
NTE5MDlaGA8yMTk5MDMyOTE1MTkwOVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
```

```
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQCjiPgW3vsXRj4JoA16WQDyoPc/eh3QBARaApJEc4nPIGoUolpAXcjFhWplo2O+
ivgfCsc4AU9OpYdAPha3spLey/bhHPRi1JZHRNqScKP0hzsCNmKhfnZTIEQCFvsp
DRp4zr91/WS06/flJFBYJ6JHhp0KwM81XQG59lV6kkoW7QIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAGLLrY3P+HH6C57dYgtJkuGZGT2+rMkk2n81/abzTJvsqRqGRrWv
XRKRXlKdM/dfiuYGokDGxiC0Mg6TYy6wvsR2qRhtXW1OtZkiHWcQCnOttz+8vpew
wx8JGMvowtuKB1iMsbwyRqZkFYLcvH+Opfb/Aayi20/ChQLdI6M2R5VU
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJAO/+DgYF78KwMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTA0Mjky
MDM1MjJaGA8yMTk4MTAwMjIwMzUyMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAv1ZLV+Z/P6INq+R1qLkzETBg7sFGKPiwHekbpuB6lrRxKHhj8V9vaReM
lnv1Ur5LAPpMPYDsuJ4WoUbPYAqVqyMAo7ikJHCCM1cXgZJefgN6z9bpS+uA3YVh
V/0ipHh/X2hc2S9wvxKWiSHu6Aq9GVpqL035tJQD+NJuqFd+nXrtcw4yGtmvA6wl
5Bjn8WdsP3xOTKjrByYY1BhXpP/f1ohU9jE9dstsRXLa+XTgTPWcWdCS2oRTWPGR
c5Aeh47nnDsyQfP9gLxHeYeQItV/BD9kU/2Hn6mnRg/B9/TYH8qzlRTzLapXp4/5
iNwusrTNexGl8BgvAPrfhjDpdgYuTwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQB7
5ya11K/hKgvaRTvZwVV8GlVZt0CGPtNvOi4AR/UN6TMm51BzUB5nurB4z0R2MoYO
Uts9sLGvSFALJ4otoB77hyNpH3drttU1CVVwal/yK/RQLSon/IoUkaGEbqalu+mH
nYad5IG4tEbmepX456XXcO58MKmnczNbPyw3FRzUZQtI/sf94qBwJ1Xo6XbzPKMy
xjL57LHIZCssD+XPifXay69OFlsCIgLim11HgPkRIHEOXLSf3dsW9r+4CjoZqB/Z
jj/P4TLCxbYCLkvglwaMjgEWF40Img0fhx7yT2X92MiSrs3oncv/IqfdVTiN8OXq
jgnq1bf+EZEZKvb6UCQV
-----END CERTIFICATE-----
```

## Eropa (Paris) — eu-west-3

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
```

```
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUaC9fX57UDr6u1vBvsCsECKBZQyIwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MzczOFoXDTI5MDQyODE2MzczOFowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUaC9fX57U
Dr6u1vBvsCsECKBZQyIwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQCARv1bQEDaMEzYI0nPlu8GHcMXgmgA94HyrXhMMcaIlQwocGBs6VILGVhM
TXP2r3JFaPEpmXSQNQHvGA13clKwAZbni8wtzv6qXb4L4muF34iQRHF0nYrEDoK7
mMPR8+oXKKuPO/mv/XKo6XAV5DDERdSYHX5kkA2R9wtvyZjPnQ==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJALWSfgHuT/ARMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNzA1MzEx
MTE4MTZaGA8yMTk2MTEwMzExMTgxNlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
```

```
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAy5V7KDqnEvF3DrSProFcgu/oL+QYD62b1U+Naq8aPuljJe127Sm9WnWA
EBdOSASkOaQ9fzjCPoG5SGgWKxYoZjsevHpmzjVv9+Ci+F57bSuMbjgUbvbRIFUB
bxQojVoXQPHgK5v433ODxkQ4sjRyUbf4YV1AFdfU7zabC698YgPVOExGhXPlTvco
8mlc631ubw2g52j0lzaozUkHPSbknTomhQIvO6kUfX0e0TDMH4jLDG2ZIrUB1L4r
OWKG4KetduFrRZyDHF6ILZu+s6ywiMicUd+2UllDFC6oas+a8D11hmO/rpWU/ieV
jj4rWAFrsebpn+Nhgy96iiVUGS2LuQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDE
iYv6FQ6knXCg+svlcaQG9q59xUC5z8HvJZ1+SxzPKKC4PKQdKvIIfE8GxVXqlZG1
cl5WKTFDMapnzb9RV/DTaVzWx3cMYT77vm1Hl1XGjhx611CGcENH1egI3lOTILsa
+KfopuJEQQ9TDMAIkGjhA+KieU/U5Ctv9fdej6d0GC6OEuwKkTNzPWue6UMq8d4H
2xqJboWsE1t4nybEosvZfQJcZ8jyIYcYBnsG13vCLM+ixjuU5MVVQNMY/gBJzqJB
V+U0QiGiuT5cYgY/QihxdHt99zwGaE0ZBC7213NKrlNuLSrqhDI2NLu8NsExqOFy
OmY0v/xVmQUQl26jJXaM
-----END CERTIFICATE-----
```

## Eropa (Spanyol) — eu-south-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC8DCCAq
+gAwIBAgIGAXjwLk46MAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAGG2m8EKmaf5qQqj3Z
+rzSaTaXE3B/R/4A2VuGqRYR7MljPtwdmU6/3CPjCACcZmTIcOAKbFiDHqadQgBZXfzGpzw8Zo
+eYmmk5fXycgnj57PYH1dIWU6I7mCbAah5MZMcmHaTmIsomGrhcnWB8d8qOU7oZ0UWK4lbiAQs1MihoUwCQYHKoZIzjg
WmbaU7YM5GwCFCvIJOesO5hZ8PHC52dAR8WWC6oe
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUbtiu7xW+c3P65VHqW/gxSFWHT8kwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI2MDYwOTA4NDMzMVoXDTMxMDYwODA4NDMzMVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
```

```
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQDB/VvR1+45Aey5zn3vPk6xBm5o9grSDL6D2iAuprQnfVXn8CIb
SDbWFhA3fi5ippjKkh3sl8VyCvCOUXKdOaNrYBrPRkrdHdBuL2Tc84RO+3m/rxIU
Z2IK1fDlC6sWAjddf6sBrV2w2a78H0H8EwuwiSgttURBjwJ7KPPJCqaqrQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUcZuHrkKFfuCOW/tfRxCPB7Ro
XQQwgZkGA1UdIwSBkTCBjoAUcZuHrkKFfuCOW/tfRxCPB7RoXQShYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUbtiu7xW+
c3P65VHqW/gxSFWHT8kwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQANhLKjQrxeWPzqlTkLK8b44iyITsV0E+o8+gJN6tKmC0Cgapp1+rCcYDzt
4R3uglYdoRDRrEPJVLjS8XcX5cefteIJRKJGm06d+fTvgzVY11IXo77tnzJF356m
Nwunpudg2r1diQCWQKKftqbdTsQxPGPVTVOFv+tXyW1s1TYj9Q==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALWSmO6DvSpQMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTgx
MzU4NDNaGA8yMjAxMTIyMjEzNTg0M1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAuAAhuSpsHC00/fD2zNlBDpNLRndi9qbHsNeuz3WqN7Samj2aSrM2hS+i
hUxx0BspZj0tZCOsbpPZ+i74NOEQtFeqQoEGvKhB1nJiF4y5I81HDhs5qHvoIivm
7rbbik3zgm1PqS/DmDjVQaXPcD31Rd9ILwBmWEwJqHigyNVlxYtCzTQcrlBrvNZM
dnNgCDAdX/HBEFxx9O12xeu0bSt0s+PJWZ1RTbYrNe7LIH6ntUqHxP/ziQ5trXEZ
uqy7aWk1L8uK4jmyNph0lbaqBa3Y6pYmU1nC27UE4i3fnPB0LSiAr+SrwVvX1g4z
ilo8kr+tbIF+JmcgYLBvO8Jwp+EUqQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUwvGzKJL9A5LReJ4Fxo5K6I20xcowgY4GA1UdIwSBhjCBg4AUwvGz
KJL9A5LReJ4Fxo5K6I20xcqhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALWSmO6DvSpQMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAJAZd31jyoTGLawAD2+v/vQsaB9vZIx5EImi
G8YGkd61uFWeNhAmtrwyE/i6FDSIphDrMHBkvw/D3BsqK+Ev/JOK/VYuaYDx/8fp
H4cwp9jC57CXzdIDREWNf6M9PsHFg2WA9XNNtClOZL5WJiJwel8eDSg+sqJUxE0l
MW+QChq/20F6niyaRK4bXrZq14as7h+F9u3A9xHEOVP7Zk9C2ehrBXzCMLSDt3GV
fEuMea2RxMhozwz34Hkdb6j18qoCfygubulovRNQjKw/cEmgPR16KfZPP5caILVt
9qkYPvePmbiVswZDee73cDymJYxLqILpOZwyXvUH8StiH42FHZQ=
-----END CERTIFICATE-----
```

## Eropa (Stockholm) — eu-north-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUN1c9U6U/xiVDFgJcYKZB4NkH1QEwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQyOTE2MDYwM1oXDTI5MDQyODE2MDYwM1owXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIEExMQ4IUN1c9U6U/
xiVDFgJcYKZB4NkH1QEwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBTIQdoFSDRHkpqNPUbZ9WXR2O5v/9bpmHojMYZb3Hw46wsaRso7STiGGX/
tRqjIkPUIXsdhZ3+7S/RmhFznmZc8e0bjU4n5vi9CJtQSt+1u4E17+V2bF+D3h/7
wcfE0l3414Q8JaTDtfEf/aF3F0uyBvr4MDMd7mFvAMmDmBPSlA==
```

```
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJALc/uRxg++EnMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xODA0MTAx
NDAwMTFaGA8yMTk3MDkxMzE0MDAxMVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAzwCGJEJIxqtr2PD2a1mA6LhRzKhTBa1AZsg3eYfpETXIVlrpojMfvVoN
qHvGshWLgrGTT6os/3gsaADheSaJKavxwX3X6tJA8fvEGqr3a1C1MffH9hBWbQqC
LbfUTAbkwis4GdTUwOwPjT1Cm3u9R/VzilCNwkj7iQ65AFAI8Enmsw3UGldEsop4
yChKB3KW3WI0FTh0+gD0YtjrqqYJxpGOYBpJp5vwdd3fZ4t1vidmDMs7liv4f9Bx
p0oSmUobU4GUlFhBchK1DukICVQdnOVzdMonYm7s+HtpFbVHR8yf6QoixBKGdSal
mBf7+y0ixjCn0pnC0VLVooGo4mi17QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQDG
4ONZiixgk2sjJctwbyD5WKLTH6+mxYcDw+3y/F0fWz561YORhP2FNnPOmEkf0Sl/
Jqk4svzJbCbQeMzRoyaya/46d7UioXMHRZam5IaGBhOdQbi97R4VsQjwQj0RmQsq
yDueDyuKTwWLK9KnvI+ZA6e6bRkdNGflK4N8GGKQ+fBhPwVELkbT9f16OJkezeeN
S+F/gDADGJgmPXfjogICb4Kvshq0H5Lm/xZlDULF2g/cYhyNY6EOI/eS5m1I7R8p
D/m6WoyZdpInxJfxW616OMkxQMRVsruLTNGtby3u1g6ScjmpFtvAMhYejBSdzKG4
FEyxIdEjoeO1jhTsck3R
-----END CERTIFICATE-----
```

## Eropa (Zürich) — eu-central-2

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjXiKJnMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAYNjaCNg/
cfgQOOl1BUj5ClUulqwZ9Q+SfDzPZh9D2C0VbiRANiZoxrV8RdgmzzC5T7VcriVwjwvta2Ch//
b+sZ86E5h0XWWr+BeEjD9cu3eDj12XB5sWEbNHNx49p5Tmtu5r2LDtlL8X/
Rpfalu2Z2OJgjFJWGf7hRwxe456n
+lowCQYHKoZIzjgEAwMvADAsAhRChsLcj4U5CVb2cp5M0RE1XbXmhAIUeGSnH+aiUQIWmPEFja+itWDufIk=
```

```
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDCzCCAnSgAwIBAgIJAK+7ZflHYRFQMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0yNjA1MjYx
MDU0MDBaFw0zMTA1MjUxMDU0MDBaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
tpnRnabOUc9o82HBjbIBE0xlLYSUa1akp1FiV05VWiG5BjyyjujqwH1ihadrIES
fLy7I6AVUeXBElSu3LitcCLY7ycov6KSmQFLrCcdoaYFN6VB4ZafQE4f1XlQ+fNo
03fSGPvLeTA2umd3xpYgchIl89CxB3YvDAWnC0RLtU8CAwEAAaOB1DCB0TALBgNV
HQ8EBAMCB4AwHQYDVR0OBBYEFDFYNKA6vm8m8PtCqgIoU4I3+EpSMIGOBgNVHSME
gYYwgYOAFDFYNKA6vm8m8PtCqgIoU4I3+EpSoWCkXjBcMQswCQYDVQQGEwJVUzEZ
MBcGA1UECBMQV2FzaGluZ3RvbiBTdGF0ZTEQMA4GA1UEBxMHU2VhdHRsZTEgMB4G
A1UEChMXQW1hem9uIFdlYiBTZXJ2aWNlcyBMTEOCCQCvu2X5R2ERUDASBgNVHRMB
Af8ECDAGAQH/AgEAMA0GCSqGSIb3DQEBCwUAA4GBADsVuZ+wv4qEfSnGx+MH8s7p
oBWc/w6DTMCltWl7+q6a7C7uTIymgfsbug31ClwB7O3dhy26Ae9gNzXfpRnWN7iK
chhz057HHESzeogH1rTHD2RkJE5z6KCC0F8JgEg+JQSSCuTFKzUuBkW4S8WMQ3Tj
kCniK/KM2bEKllCqvINx
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJALvT012pxTxNMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTgx
NTEyMDdaGA8yMjAxMTIyMjE1MTIwN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAyn+Lsnq1ykrfYlZkk6aAAYNReNd9Iw8AUwCBkgOr2eBiBBepYxHwU85N
++moQ+j0EV2VaahBeTLShGZZS1HsyK8+cYT2QzpghIoamcYhrPXyIxlWiRQlaqSg
OFiE9bsqL3rCF5Vz+tOiTe5W/7ojfOFls6++g7ZpobwJlpMbuJepqyeHMPyjvO5A
age81lJewc4bxo2ntaW0HCqNksqfYB78j6X6kn3PFpX7FaYAwZA+Xx6C7UCY7rNi
UdQzfAo8htfJi4chz7frpUdQ9kl3IOQrsLshBB5fFUjl09NiFipCGBwi+8ZMeSn1
5qwBI01BWXPfG7WX6OwyjhmH6JtE1wIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU8HN4vvJrsZgPQeksMBgJb9xR1yYwgY4GA1UdIwSBhjCBg4AU8HN4
vvJrsZgPQeksMBgJb9xR1yahYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJALvT012pxTxNMBIGA1UdEwEB/wQIMAYBAf8C
```

```
AQAwDQYJKoZIhvcNAQELBQADggEBAGlHYDtcHpfBvdHx9HeQE8HgNugJUPdEqxun
t9U33p8VFrs+uLPtrOd9HDJEGvvs5h84EUie/oGJxRt7V1Vlid1PvHf6cRmpjgqY
YdggAVkZtY/PnFVmzf2bMVlSQPrqCl7U0zaw2Kvnj4zgX0rZyCetgrRZSUSxotyp
978Wy9ccXwVSeYG/YAr5rJpS6ZH7eRQvUY0IzwFNeaOPgOTEVpcjWlV6+MQEvsEx
W85q+s6AVr49eppEx8SLJsl0C23yB+L+t32tAveQImRWtJMpzZ5cxh/sYgDVeoC0
85H1NK/7H9fAzT1cPu1oHSnB0xYzzHGOAmXmusMfwUk8fL1RQkE=
-----END CERTIFICATE-----
```

## Israel (Tel Aviv) — il-central-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq+gAwIBAgIGAX0QPi
+9MAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcMB1NlYX
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/
hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAbazCL5XXyPmcw3+oMYQUF5/9YogW6D0FZbYuyPgjOoUwWdl6fjlzWca
pq+l1ezuK2DFOzNTEyPEwwCQYHKoZIzjgEAwMvADAsAhRt1jkpXsvrS
+xTo2M9h2s2uLAhEQIUOZ2FcnTSrshF2EIdixZZwtNv66Q=
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUPgjnsnqT7OaGnZgGF7/43ihPh+0wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI2MDYxMjA5MjQyOVoXDTMxMDYxMTA5MjQyOVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQDrc24u3AgFxnoPgzxR6yFXOamcPuxYXhYKWmapb+S8vOy5hpLo
Re4RkOrY0cM3bN07GdEMlin5mU0y1t8y3ct4YewvmkgT42kTyMM+t1K4S0xsqjXx
xS716uGYh7eWtkxrCihj8AbXN/6pa095h+7TZyl2n83keiNUzM2KoqQVMwIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQU+dSlCYS89nZaff8nPOdiQJje
2PkwgZkGA1UdIwSBkTCBjoAU+dSlCYS89nZaff8nPOdiQJje2PmhYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUPgjnsnqT
```

```
7OaGnZgGF7/43ihPh+0wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQDOq6yTQkIXzAZffpLDBss1R8VZzRoLvRzsFa1xpOcRnPqIm0untEHsOGX/
q34FiGQBOVY6btNaZb7aQXDlZGX+sGvst6BYC2u6H6Zpyo0qy0mz6AGA3eVEonJ2
lKpvAwBkrGajRC8JW0Gx31UZDmAcn3gjQi5zMWFead9mJJJA3g==
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAOVp1h2I9wW7MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA3MTUx
MjQ0MTJaGA8yMjAxMTIxOTEyNDQxMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA13PkyWvl61iV/SYfO1UFO76UpDfPm2SF/Rz/o33cm699X++EYPxTnoEc
vmWeS0I7eDXc4OCUiToG0sEx0klE0CX1Z1tK6qJ+zgWQLZ9SZEC9H0NsSA6LhrHu
Nq0dzeK3LjhdfcX46/4GqdiptpdTuM4m/hOQ5yx4JMQ/n1sdpv4M5VLRWwWW9Lem
ufb79Id7O9SispxgRsz1KXIjp7N9S4BY7itSXz97uSyzTqEjWZ6mDUhTu3t21GKC
6flALGTTrG2yghEhz53rkvLsvwzjPSSlT6LIf0mrRPzHaf+EdaKoasElE1SHh+ZH
9mI81HywpE+HZ+W+5hBCvjYp90Y1fwIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQU58tN2JO+yEGq5JbIXxGi4vRVPyIwgY4GA1UdIwSBhjCBg4AU58tN
2JO+yEGq5JbIXxGi4vRVPyKhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAOVp1h2I9wW7MBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBANBN0e1EqNy4+IU2yQzMJ+Wy5ZIOtTP6GSBR
7muVY1bDeAwtNTEOpwgrZVlC7/xq5Q0LC1y0Z70hHXEf8au7qStaAoUtxzvhTAZI
NC0lwoFU56UFw4N0vZII17iqEfoqRC4PpI3OxqEJHFy0VLlvAzJoKB4QLLqDAYVA
LXCiOLoVT+y9tRYsxw5MyO0Bi6fxQIIADl2bE9xkunTN1Jkkwqo3LxNy/ryz4QWR
8K7jHUItifv4h/hxBKpHEquN8CkdvM9oeGl7I8PFrSFEpGr1euDXY0euZzzYiDBV
m6GpTJgzpVsEuIX52dPcPemwQncoIfZyhWDW85MJUnby2WTEcFo=
-----END CERTIFICATE-----
```

## Meksiko (Tengah) — mx-central-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq8CCQD4QwfTExrgxTAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDExMTQwOTQ4MjRaGA8y
MDUwMTExNTA5NDgyNFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0
```

```
b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIg
U2VydmljZXMgTExDMIIBtzCCASwGByqGSM44BAEwggEfAoGBAK+pJEOtD2RFlCbf
oAzONffZa42FB8G6OA/30DT81xWvB2Hnmgw7LfiELYqgfuck3YSDffNDcmiFTJuX
AU9u0NtbbxcOytcdUq4HInpHiB85I6WWWKYB66aGbvYowUPpqBZkASb1i/pYAh5W
nNYvx408008tTzqpMfZDchJHzWqBAhUA0ogoJzTw2/4pKhz9aqTsRCzRVPsCgYEA
qklRUcuU0du/bT/M6kWxYvrTGhO9KXQe+7RtbaIq4dWRsCBnO4smDY/GmI9H8ew1
LRJ9AcLGmxDm795CvVZkHNcht7PDAREagWmz2YvhLA+ev6U0RpfdlBXCck2p1CxQ
LMtoFO7DThksHIQBtnlQdHDvguKEZQz/Iobhne6Kb3wDgYQAAoGAQSTlOqFq6RtR
Jvp406XhGl+eO9SQUpevpzG3Dzbdy6EQ8PBJDOHHJvxbebvpnQssh0CnQfTkSw26
jwPy9V5zRC0ZezwnRTyGSJwiErUKDGeVekEAoNJL1USy8jKkgBajlskZrR+0JHDN
Jv3UwYIPplc4ZS2f2E7btrtlaWt/P70wCQYHKoZIzjgEAwMvADAsAhRXljPWKWYf
61DkDVdgPPHsS2/LuwIUejcGV9WuS3uPvJ6lmn4opxUGBZw=
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIICPzCCAaigAwIBAgIUCmzpTTMBQYItpMC2VDYsZfIAS7IwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcMB1NlYXR0bGUxIDAeBgNVBAoMF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDQwMjEyNTAzNloXDTI5MDQwMTEyNTAzNlowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcMB1NlYXR0bGUxIDAe
BgNVBAoMF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQDfvOnCzm1iN58Nm7k6ehoy6v0lnnFI617D6CY3bfuq01RCdEQL
96+pYawJieTH8JAQKj02CAa3AeaqdXTE/pDhI/YKLreeMb4K68WMn24Wjjs6oxjB
bAmsKXtt9ihKHGBFNUhgFrNFYyA2i7ieJviwpHjQ/XgXiG2u1/t/4VydUwIDAQAB
MA0GCSqGSIb3DQEBCwUAA4GBAL5+vvj4lhaE+J5tuCqV3XJzDd97lsD4le2O2uGw
P0sGdUcRAdxzU3Bwq/hhtzNWnfwo0aCEQKmLM7xyd3nUa0VvKXLq+DDuayipWINr
OATnNxFRe99d38qHTR1dgkjZdKbbtnl6O4fgM57tVEuQJd/N4ILl9jaRcJ5Ip+9t
3y5t
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJALxuE00HoJomMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDEwMzEx
MTE0MzNaGA8yMjA0MDQwNjExMTQzM1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEA13xWucALo8M/TXbZJgHrqFqFON91dSLPE/eLbmeIJbP1pb9ICd33qKAx
HlKSrXI9b9YS3U1Pl0bF3ZgfeE/x4Y0KDDZwzpf07H8IgRittULJoNLYVKCJXWPq
```

```
Ky1qvDJX3653dUbUu9eAdVCTRgk7eKpPBLAmW27+pgAGzEYrVV3u2AvqNTonvfTU
sPgEVnAL1J490pNM85KtFynxFTWGigHkd3BHidxmLrTH4I4eRxNZ9q/3gsDW+zKt
jQlpM7JzZa2OqxsF5YQDh1fF52Emqsr+ufLeGqDLOgT1QWcqpz57AX8GqZpgZULo
itCRNXbQDzZY9FxIgpiFJv3y/qYYDQIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQCt
MTeH4EgqjJEjb1qm5tzXYurLprVrEVQ+PhGXJFJd3xAZyeaDVYy7kio08E2xhmHd
HtkBDty2KnOHsTQmeAJCci7d4tYXZ/1qe341wwm9OoFcO8jhIndx6FXJCgQUY4dL
AAr9HQJFWG5dMZgbi1Zuhxdio3sSoOBjL2p7QIsGNkITvCDIs/H0/szpJnyyyIqu
wmUhSel5hdy5MwOsyUKVGnAdaS5Vd9oL4kLszS9nBZ7ny6BC9odIkFAdGqQ5vM4z
vcbfOq14hjatQmJgJhksN/0Dpl78Gheq0pIhP8LTkAOEG2832nQLzCa3oxSk8otG
GJXkzzyQjse+13r8+yNJ
-----END CERTIFICATE-----
```

## Timur Tengah (Bahrain) — me-south-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7jCCAq4CCQCVWIgSmP8RhTAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xOTAyMDUxMzA2MjFaFw00
NTAyMDUxMzA2MjFaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbgwggEsBgcqhkjOOAQBMIIBHwKBgQDcwojQfgWdV1QliO0B
8n6cLZ38VE7ZmrjZ9OQV//Gst6S1h7euhC23YppKXi1zovefSDwFU54zi3/oJ++q
PHlP1WGL8IZ34BUgRTtG4TVolvp0smjkMvyRu5hIdKtzjV93Ccx15gVgyk+o1IEG
fZ2Kbw/Dd8JfoPS7KaSCmJKxXQIVAIZbIaDFRGa2qcMkW2HWASyNDl7bAoGBANtz
IdhfMq+l2I5iofY2oj3HI21Kj3LtZrWEg3W+/4rVhL3lTm0Nne1rl9yGujrjQwy5
Zp9V4A/w9w2O10Lx4K6hj34Eefy/aQnZwNdNhv/FQP7Az0fju+Yl6L13OOHQrL0z
Q+9cF7zEosekEnBQx3v6psNknKgD3Shgx+GO/LpCA4GFAAKBgQCVS7m77nuNAlZ8
wvUqcooxXMPkxJFl54NxAsAul9KP9KN4svm0O3Zrb7t2FOtXRM8zU3TqMpryq1o5
mpMPsZDg6RXo9BF7Hn0DoZ6PJTamkFA6md+NyTJWJKvXC7iJ8fGDBJqTciUHuCKr
12AztQ8bFWsrTgTzPE3p6U5ckcgV1TAJBgcqhkjOOAQDAy8AMCwCFB2NZGWm5EDl
86ayV3c1PEDukgQIAhQow38rQkN/VwHVeSW9DqEshXHjuQ==
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDPDCCAqWgAwIBAgIJAMl6uIV/zqJFMA0GCSqGSIb3DQEBCwUAMHIxCzAJBgNV
BAYTAlVTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAdTZWF0dGxlMSAw
HgYDVQQKDBdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzEaMBgGA1UEAwwRZWMyLmFt
YXpvbmF3cy5jb20wIBcNMTkwNDI2MTQzMjQ3WhgPMjE5ODA5MjkxNDMyNDdaMHIx
```

```
CzAJBgNVBAYTAlVTMRMwEQYDVQQIDApXYXNoaW5ndG9uMRAwDgYDVQQHDAdTZWF0
dGxlMSAwHgYDVQQKDBdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzEaMBgGA1UEAwwR
ZWMyLmFtYXpvbmF3cy5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBALVN
CDTZEnIeoX1SEYqq6k1BV0ZlpY5y3KnoOreCAE589TwS4MX5+8Fzd6AmACmugeBP
Qk7Hm6b2+g/d4tWycyxLaQlcq81DB1GmXehRkZRgGeRge1ePWd1TUA0I8P/QBT7S
gUePm/kANSFU+P7s7u1NNl+vynyi0wUUrw7/wIZTAgMBAAGjgdcwgdQwHQYDVR0O
BBYEFILtMd+T4YgH1cgc+hVsVOV+480FMIGkBgNVHSMEgZwwgZmAFILtMd+T4YgH
1cgc+hVsVOV+480FoXakdDByMQswCQYDVQQGEwJVUzETMBEGA1UECAwKV2FzaGlu
Z3RvbjEQMA4GA1UEBwwHU2VhdHRsZTEgMB4GA1UECgwXQW1hem9uIFdlYiBTZXJ2
aWNlcyBMTEMxGjAYBgNVBAMMEWVjMi5hbWF6b25hd3MuY29tggkAyXq4hX/OokUw
DAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQsFAAOBgQBhkNTBIFgWFd+ZhC/LhRUY
4OjEiykmbEp6hlzQ79T0Tfbn5A4NYDI2icBP0+hmf6qSnIhwJF6typyd1yPK5Fqt
NTpxxcXmUKquX+pHmIkK1LKDO8rNE84jqxrxRsfDi6by82fjVYf2pgjJW8R1FAw+
mL5WQRFexbfB5aXhcMo0AA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANZkFlQR2rKqMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xOTAyMDUx
MzA2MjBaGA8yMTk4MDcxMTEzMDYyMFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAy4Vnit2eBpEjKgOKBmyupJzJAiT4fr74tuGJNwwa+Is2vH12jMZn9Il1
UpvvEUYTIboIgISpf6SJ5LmV5rCv4jT4a1Wm0kjfNbiIlkUi8SxZrPypcw24m6ke
BVuxQZrZDs+xDUYIZifTmdgD50u5YE+TLg+YmXKnVgxBU6WZjbuK2INohi71aPBw
2zWUR7Gr/ggIpf635JLU3KIBLNEmrkXCVSnDFlsK4eeCrB7+UNak+4BwgpuykSGG
Op9+2vsuNqFeU1l9daQeG9roHR+4rIWSPa0opmMxv5nctgypOrE6zKXx2dNXQldd
VULv+WH7s6Vm4+yBeG8ctPYH5GOo+QIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBs
ZcViiZdFdpcXESZP/KmZNDxB/kktlIEIhsQ+MNn29jayE5oLmtGjHj5dtA3XNKlr
f6PVygVTKbtQLQqunRT83e8+7iCZMKI5ev7pITUQVvTUwI+Fc01JkYZxRFlVBuFA
WGZO+98kxCS4n6tTwVt+nSuJr9BJRVC17apfHBgSS8c5OWna0VU/Cc9ka4eAfQR4
7pYSDU3wSRE01cs30q34lXZ629IyFirSJ5TTOIc0osNL7vwMQYj8HOn4OBYqxKy8
ZJyvfXsIPh0Na76PaBIs6ZlqAOflLrjGzxBPiwRM/XrGmF8ze4KzoUqJEnK13O6A
KHKgfiigQZ1+gv5FlyXH
-----END CERTIFICATE-----
```

### Timur Tengah (UEA) — me-central-1

#### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7zCCAq
+gAwIBAgIGAXjXhqnnMAkGByqGSM44BAMwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgMEFdhc2hpbmd0b24gU3RhdGUxED
U4EddRIpUt9KnC7s5Of2EbdSPO9EAMMeP4C2USZpRV1AIlH7WT2NWPq/
xfW6MPbLm1Vs14E7gB00b/JmYLdrmVClpJ+f6AR7ECLCT7up1/63xhv4O1fnxqimFQ8E
+4P208UewwI1VBNaFpEy9nXzrith1yrv8iIDGZ3RSAHHAhUAl2BQjxUjC8yykrmCouuEC/
BYHPUCgYEA9+GghdabPd7LvKtcNrhXuXmUr7v6OuqC+VdMCz0HgmdRWVeOutRZT
+ZxBxCBgLRJFnEj6EwoFhO3zwkyjMim4TwWeotUfI0o4KOuHiuzpnWRbqN/C/ohNWLx
+2J6ASQ7zKTxvqhRkImog9/hWuWfBpKLZl6Ae1UlZAFMO/7PSSoDgYQAAoGAW+csuHsWp/7/
pv8CTKFwxsYudxuR6rbWaHCykIeAydXL9AWnphK6yp1ODEMBFl68Xq8Hp23sOWyf8moOhqCom9+0+ovuUFdpvCie86bp
TOZU568Ty1ff3dDWbdRzeNQRHodRG+XEQSizMkAreeWt4kBa+PUwCQYHKoZIzjgEAwMvADAsAhQD3Z
+XGmzKmgaLgGcVX/Qf1+Tn4QIUH1cgksBSVKbWj81tovBMJeKgdYo=
-----END CERTIFICATE-----
```

#### RSA

```
-----BEGIN CERTIFICATE-----
MIICMzCCAZygAwIBAgIGAXjRrnDjMA0GCSqGSIb3DQEBBQUAMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIDBBXYXNoaW5n
KyA6zyruJQrYy00a6wqLA7eeUzk3bMiTkLsTeDQfrkaZMfBAjGaaOymRo1C3qzE4rIenmahvUplu9ZmLwL1idWXMRX2R
+d2SeoKOKQWoc2UOFZMHYxDue7zkyk1CIRaBukTeY13/
RIrlc6X61zJ5BBtZXlHwayjQIDAQABMA0GCSqGSIb3DQEBBQUAA4GBABTqTy3R6RXKPW45FA+cgo7YZEj/
Cnz5YaoUivRRdX2A83BHuBTvJE2+WX00FTEj4hRVjameE1nENoO8Z7fUVloAFDlDo69fhkJeSvn51D1WRrPnoWGgEfr1
B+Wqm3kVEz/QNcz6npmA6
-----END CERTIFICATE-----
```

#### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAM4h7b1CVhqqMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yMjA0MTEx
MDE1MDNaGA8yMjAxMDkxNTEwMTUwM1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEApybTWfMOhSoMpqPo72eqAmnn1dXGZM+G8EoZXzwHwT/+IHEXNB4q5N6k
tudYLre1bJxuzEw+iProSHjmb9bB9YscRTofjVhBlt35Fc+i8BaMeH94SR/eE8QO
mll8gnLNW3d62lyuhzuyv1e5wVlRqzYw+X2zRH4/wRDOCOpzjKoHIgyPKsMgwsw5
aTZhNMsGxZN9dbkf0iCGeQLDytwU/JTh/HqvSr3VfUOapTJJiyAxoCtZWgp1/7wC
RvOCSMRJobpUqxZgl/VsttwNkikSFz1wGkcYeSQvk+odbnYQckA8tdddoVI56eD4
```

```
qtREQvfpMAX5v7fcqLexl5d5vH8uZQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUOadrbTs+OhzwoAgUJ7RqQNdwufkwgY4GA1UdIwSBhjCBg4AUOadr
bTs+OhzwoAgUJ7RqQNdwufmhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAM4h7b1CVhqqMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAICTdAOGEOnII8HaGCpCB8us/hGFaLptJaAf
D5SJAyVy66/mdfjGzE1BKkKxnbxemEVUIzbRidOnyilB+pKwN3edAjTZtWdpVA0V
R/G/qQPmcVljtycBz4VC6SuOUYflGzLH1GZ6GJWbuDtFzw8r7HGdRN1wrEPe3UF2
sMpuVezqnRUdvVRoVQP4jFgNsE7kNvtn2NiPhb/CtrxpcwIQ7r6YeoHcBSheuV1Z
xZDHynC3KUprQGx1+Z9QqPrDf180MaoqAlTl4+W6Pr2NJYrVUFGS/ivYshMg574l
CPU6r4wWZSKwEUXq4BInYX6z6iclp/p/J5QnJp2mAwyi6M+I13Y=
-----END CERTIFICATE-----
```

## Amerika Selatan (Sao Paulo) — sa-east-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUX4Bh4MQ86Roh37VDRRX1MNOB3TcwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
```

```
MB4XDTI0MDQyOTE2NDYwOVoXDTI5MDQyODE2NDYwOVowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCHvRjf/0kStpJ248khtIaN8qkDN3tkw4VjvA9nvPl2anJO+eIB
UqPfQG09kZlwpWpmyO8bGB2RWqWxCwuB/dcnIob6w420k9WY5C0IIGtDRNauN3ku
vGXkw3HEnF0EjYr0pcyWUvByWY4KswZV42X7Y7XSS13hOIcL6NLA+H94/QIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQUJdbMCBXKtvCcWdwUUizvtUF2
UTgwgZkGA1UdIwSBkTCBjoAUJdbMCBXKtvCcWdwUUizvtUF2UTihYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUX4Bh4MQ8
6Roh37VDRRX1MNOB3TcwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBnhocfH6ZIX6F5K9+Y9V4HFk8vSaaKL5ytw/P5td1h9ej94KF3xkZ5fyjN
URvGQv3kNmNJBoNarcP9I7JIMjsNPmVzqWawyCEGCZImoARxSS3Fc5EAs2PyBfcD
9nCtzMTaKO09Xyq0wqXVYn1xJsE5d5yBDsGrzaTHKjxo61+ezQ==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAMcyoxx4U0xxMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA4MTQw
ODU4MDJaGA8yMTk1MDExNzA4NTgwMlowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAw45IhGZVbQcy1fHBqzROhO8CsrDzxj/WP4cRbJo/2DAnimVrCCDs5O86
FA39Zo1xsDuJHDlwMKqeXYXkJXHYbcPWc6EYYAnR+PlLG+aNSOGUzsy202S03hT0
B20hWPCqpPp39itIRhG4id6nbNRJOzLm6evHuepMAHR4/OV7hyGOiGaV/v9zqiNA
pMCLhbh2xk0PO35HCVBuWt3HUjsgeks2eEsu9Ws6H3JXTCfiqp0TjyRWapM29OhA
cRJfJ/d/+wBTz1fkWOZ7TF+EWRIN5ITEadlDTPnF1r8kBRuDcS/lIGFwrOOHLo4C
cKoNgXkhTqDDBDu6oNBb2rS0K+sz3QIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUqBy7D847Ya/w321Dfr+rBJGsGTwwgY4GA1UdIwSBhjCBg4AUqBy7
D847Ya/w321Dfr+rBJGsGTyhYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAMcyoxx4U0xxMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBACOoWSBf7b9AlcNrl4lr3QWWSc7k90/tUZal
PlT0G3Obl2x9T/ZiBsQpbUvs0lfotG0XqGVVHcIxF38EbVwbw9KJGXbGSCJSEJkW
vGCtc/jYMHXfhx67Szmftm/MTYNvnzsyQQ3v8y3Rdah+xe1NPdpFrwmfL6xe3pFF
cY33KdHA/3PNLdn9CaEsHmcmj3ctaaXLFIzZhQyyjtsrgGfTLvXeXRokktvsLDS/
YgKedQ+jFjzVJqgr4NjfY/Wt7/8kbbdhzaqlB5pCPjLLzv0zp/XmO6k+JvOePOGh
JzGk5t1QrSju+MqNPFk3+1O7o910Vrhqw1QRB0gr1ExrviLbyfU=
-----END CERTIFICATE-----
```

## AWS GovCloud (US-East) - us-gov-east-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIULVyrqjjwZ461qelPCiShB1KCCj4wDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDUwNzE1MjIzNloXDTI5MDUwNjE1MjIzNlowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCpohwYUVPH9I7Vbkb3WMe/JB0Y/bmfVj3VpcK445YBRO9K80al
esjgBc2tAX4KYg4Lht4EBKccLHTzaNi51YEGX1aLNrSmxhz1+WtzNLNUsyY3zD9z
vwX/3k1+JB2dRA+m+Cpwx4mjzZyAeQtHtegVaAytkmqtxQrSCexBxvqRqQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQU1ZXneBYnPvYXkHVlVjg7918V
gE8wgZkGA1UdIwSBkTCBjoAU1ZXneBYnPvYXkHVlVjg7918VgE+hYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IULVyrqjjw
Z461qelPCiShB1KCCj4wEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQBfAL/YZv0y3zmVbXjyxQCsDloeDCJjFKIu3ameEckeIWJbST9LMto0zViZ
puIAf05x6GQiEqfBMk+YMxJfcTmJB4Ebaj4egFlslJPSHyC2xuydHlr3B04INOH5
Z2oCM68u6GGbj0jZjg7GJonkReG9N72kDva/ukwZKgq8zErQVQ==
```

```
-----END CERTIFICATE-----
```

### RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJALPB6hxFhay8MA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xODA0MTAx
MjMyNDlaGA8yMTk3MDkxMzEyMzI0OVowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAva9xsI9237KYb/SPWmeCVzi7giKNron8hoRDwlwwMC9+uHPd53UxzKLb
pTgtJWAPkZVxEdl2Gdhwr3SULoKcKmkqE6ltVFrVuPT33La1UufguT9k8ZDDuO9C
hQNHUdSVEuVrK3bLjaSsMOS7Uxmnn7lYT990IReowvnBNBsBlcabfQTBV04xfUG0
/m0XUiUFjOxDBqbNzkEIblW7vK7ydSJtFMSljga54UAVXibQt9EAIF7B8k9l2iLa
mu9yEjyQy+ZQICTuAvPUEWe6va2CHVY9gYQLA31/zU0VBKZPTNExjaqK4j8bKs1/
7dOV1so39sIGBz21cUBec1o+yCS5SwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQBt
hO2W/Lm+Nk0qsXW6mqQFsAou0cASc/vtGNCyBfoFNX6aKXsVCHxq2aq2TUKWENs+
mKmYu1lZVhBOmLshyllh3RRoL3Ohp3jCwXytkWQ7ElcGjDzNGc0FArzB8xFyQNdK
MNvXDi/ErzgrHGSpcvmGHiOhMf3UzChMWbIr6udoDlMbSIO7+8F+jUJkh4Xl1lKb
YeN5fsLZp7T/6YvbFSPpmbn1YoE2vKtuGKxObRrhU3h4JHdp1Zel1pZ6lh5iM0ec
SD11SximGIYCjfZpRqI3q50mbxCd7ckULz+UUPwLrfOds4VrVVSj+x0ZdY19Plv2
9shw5ez6Cn7E3IfzqNHO
-----END CERTIFICATE-----
```

## AWS GovCloud (US-West) - us-gov-west-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC7TCCAq0CCQCWukjZ5V4aZzAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAeFw0xMjAxMDUxMjU2MTJaFw0z
ODAxMDUxMjU2MTJaMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9u
IFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNl
cnZpY2VzIExMQzCCAbcwggEsBgcqhkjOOAQBMIIBHwKBgQCjkvcS2bb1VQ4yt/5e
ih5OO6kK/n1Lzllr7D8ZwtQP8fOEpp5E2ng+D6Ud1Z1gYipr58Kj3nssSNpI6bX3
VyIQzK7wLclnd/YozqNNmgIyZecN7EglK9ITHJLP+x8FtUpt3QbyYXJdmVMegN6P
hviYt5JH/nYl4hh3Pa1HJdskgQIVALVJ3ER11+Ko4tP6nwvHwh6+ERYRAoGBAI1j
k+tkqMVHuAFcvAGKocTgsjJem6/5qomzJuKDmbJNu9Qxw3rAotXau8Qe+MBcJl/U
hhy1KHVpCGl9fueQ2s6IL0CaO/buycU1CiYQk40KNHCcHfNiZbdlx1E9rpUp7bnF
lRa2v1ntMX3caRVDdbtPEWmdxSCYsYFDk4mZrOLBA4GEAAKBgEbmeve5f8LIE/Gf
```

```
MNmP9CM5eovQOGx5ho8WqD+aTebs+k2tn92BBPqeZqpWRa5P/+jrdKml1qx4llHW
MXrs3IgIb6+hUIB+S8dz8/mmO0bpr76RoZVCXYab2CZedFut7qc3WUH9+EUAH5mw
vSeDCOUMYQR7R9LINYwouHIziqQYMAkGByqGSM44BAMDLwAwLAIUWXBlk40xTwSw
7HX32MxXYruse9ACFBNGmdX2ZBrVNGrN9N2f6ROk0k9K
-----END CERTIFICATE-----
```

## RSA

```
-----BEGIN CERTIFICATE-----
MIIDITCCAoqgAwIBAgIUe5wGF3jfb7lUHzvDxmM/ktGCLwwwDQYJKoZIhvcNAQEL
BQAwXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAO
BgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExD
MB4XDTI0MDUwNzE3MzAzMloXDTI5MDUwNjE3MzAzMlowXDELMAkGA1UEBhMCVVMx
GTAXBgNVBAgTEFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAe
BgNVBAoTF0FtYXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUA
A4GNADCBiQKBgQCpohwYUVPH9I7Vbkb3WMe/JB0Y/bmfVj3VpcK445YBRO9K80al
esjgBc2tAX4KYg4Lht4EBKccLHTzaNi51YEGX1aLNrSmxhz1+WtzNLNUsyY3zD9z
vwX/3k1+JB2dRA+m+Cpwx4mjzZyAeQtHtegVaAytkmqtxQrSCexBxvqRqQIDAQAB
o4HfMIHcMAsGA1UdDwQEAwIHgDAdBgNVHQ4EFgQU1ZXneBYnPvYXkHVlVjg7918V
gE8wgZkGA1UdIwSBkTCBjoAU1ZXneBYnPvYXkHVlVjg7918VgE+hYKReMFwxCzAJ
BgNVBAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdT
ZWF0dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQ4IUe5wGF3jf
b7lUHzvDxmM/ktGCLwwwEgYDVR0TAQH/BAgwBgEB/wIBADANBgkqhkiG9w0BAQsF
AAOBgQCbTdpx1Iob9SwUReY4exMnlwQlmkTLyA8tYGWzchCJOJJEPfsW0ryy1A0H
YIuvyUty3rJdp9ib8h3GZR71BkZnNddHhy06kPs4p8ewF8+d8OWtOJQcI+ZnFfG4
KyM4rUsBrljpG2aOCm12iACEyrvgJJrS8VZwUDZS6mZEnn/lhA==
-----END CERTIFICATE-----
```

## RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIDOzCCAiOgAwIBAgIJANCOF0Q6ohnuMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0xNTA5MTAx
OTQyNDdaGA8yMTk1MDIxMzE5NDI0N1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAzIcGTzNqie3f1olrrqcfzGfbymSM2QfbTzDIOG6xXXeFrCDAmOq0wUhi
3fRCuoeHlKOWAPu76B9os71+zgF22dIDEVkpqHCjBrGzDQZXXUwOzhm+PmBUI8Z1
qvbVD4ZYhjCujWWzrsX6Z4yEK7PEFjtf4M4W8euw0RmiNwjy+knIFa+VxK6aQv94
lW98URFP2fD84xedHp6ozZlr3+RZSIFZsOiyxYsgiwTbesRMI0Y7LnkKGCIHQ/XJ
OwSISWaCddbu59BZeADnyhl4f+pWaSQpQQ1DpXvZAVBYvCH97J1oAxLfH8xcwgSQ
/se3wtn095VBt5b7qTVjOvy6vKZazwIDAQABMA0GCSqGSIb3DQEBCwUAA4IBAQA/
```

```
S8+a9csfASkdtQUOLsBynAbsBCH9Gykq2m8JS7YE4TGvqlpnWehz78rFTzQwmz4D
fwq8byPkl6DjdF9utqZ0JUo/Fxelxom0h6oievtBlSkmZJNbgc2WYm1zi6ptViup
Y+4S2+vWZyg/X1PXD7wyRWuETmykk73uEyeWFBYKCHWsO9sI+62O4Vf8Jkuj/cie
1NSJX8fkervfLrZSHBYhxLbL+actVEo00tiyZz8GnhgWx5faCY38D/k4Y/j5Vz99
7lUX/+fWHT3+lTL8ZZK7fOQWh6NQpI0wTP9KtWqfOUwMIbgFQPoxkP00TWRmdmPz
WOwTObEf9ouTnjG9OZ20
-----END CERTIFICATE-----
```

## AWS Awan Berdaulat Eropa — eusc-de-east-1

### DSA

```
-----BEGIN CERTIFICATE-----
MIIC8DCCAq8CCQDyckpYH0omIDAJBgcqhkjOOAQDMFwxCzAJBgNVBAYTAlVTMRkw
FwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYD
VQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDEwMTgwOTM5MzdaGA8y
MDUwMTAxOTA5MzkzN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgTEFdhc2hpbmd0
b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0FtYXpvbiBXZWIg
U2VydmljZXMgTExDMIIBtzCCASsGByqGSM44BAEwggEeAoGBAJR1zx7FzSGhXd3m
U1PfwH33QEK4CHnFSnp/DUse0SdLwsvLkHw7Tq3M7IX4E+TZ6OzYZZeoVZttqbyq
OZ8ROUS7CzDWcQ0503de/ya51cTs4BiO4nv2awm1WEoV9XtQyA2JLn0Jv7g2uja0
317krS+Dk3kjs9RpWvkmiyTMu1YhAhUA69U+A26BWLVpNIZZ0uWMZJSNlIkCgYBw
w1cgrHFGEqJNhqN7Sjs2YwTP+jpp1GvhxKyce5aOh5ErVvufb5gVVoKahrrHEBms
TrBXxnfcPaTupwdmEZ9eM8a+9RS2vye//KU1kGIOMgafPEzBEXs9m7okmAQpYXnG
cVOTtx2gd/E/VfKRiMSAmam178qwvcbcijNqRdLjRQOBhQACgYEAiEFVGYVCntmy
va8c2oU5/owdMws+NZ6u9blat0ymPr/8Q+DR77CgUPJW+K0k41GdgWpVlrsjiiPw
iygzw4DW7Vsl018KcIFUTiSwpzv2in+tKNUkO5Ly0q6arbSxrXZffYe7aOZKtCCH
XWDMKOH2iCvZptMpG7fDDSc7Be8v4AgwCQYHKoZIzjgEAwMwADAtAhQyvCF3q437
RvdPXMFcMc6qrD0svAIVAMvsN3NwOY4KRYlFv9Dnok8D/ixI
-----END CERTIFICATE-----
```

### RSA

```
-----BEGIN CERTIFICATE-----
MIICNjCCAZ+gAwIBAgIJAPybReXZqUSNMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNDEwMTgw
OTM5MzdaGA8yMjA0MDMyNDA5MzkzN1owXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKB
gQDShbskmohorgcjN7gfApw89CEsg7praCDggaUVXF6gy9JdJ5INcv7+DkwKaAxd
NQWDsWZnxIOqPuPfyAYVP3D2/exP3Rfxj31UomLduu7S83zgiJ8VVL/GzlDELv5/
```

```
K2rTcd+580NjLSa6FYDcXDip2ir6TXxRYBYHP65dwniE+wIDAQABMA0GCSqGSIb3
DQEBCwUAA4GBAKu9t3CAtZt7c+k4bCZlmxrwgps1668p24x7QRmLxgXnwj5A2MOM
qEtxA9Q3q3WxUc0nWKaCTvRrTfWW917xsLcvVL8rI6xCjg2bb5FJ73xzcfSxHr9I
Q1mThJgLaWVIblllGJK7+EogR7gfqlPhc0Ql5bYuaozdt+r69xNuJlyR
-----END CERTIFICATE-----
```

RSA-2048

```
-----BEGIN CERTIFICATE-----
MIIEEjCCAvqgAwIBAgIJAODRxRi5VXBbMA0GCSqGSIb3DQEBCwUAMFwxCzAJBgNV
BAYTAlVTMRkwFwYDVQQIExBXYXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0
dGxlMSAwHgYDVQQKExdBbWF6b24gV2ViIFNlcnZpY2VzIExMQzAgFw0yNTAyMjEx
MjQ0MTRaGA8yMjA0MDcyODEyNDQxNFowXDELMAkGA1UEBhMCVVMxGTAXBgNVBAgT
EFdhc2hpbmd0b24gU3RhdGUxEDAOBgNVBAcTB1NlYXR0bGUxIDAeBgNVBAoTF0Ft
YXpvbiBXZWIgU2VydmljZXMgTExDMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAr24OkDrpRU1ffx1PAcHZbKu7fZkhfT/e/PDhSkp2FvOzJEqDUT23E2wp
K/44mhJ3d8blYtgBiRXjqRsAFu8ZbzXLc0gaQ9A5woxfx5b7iOMwneCajS9xS5qr
cE0MMEY9XwG8MCp9YBoDJqDJh74oQqcPROBq16L4k2Ys22n1khus4JpwT7VSRE1I
nnt7fGIYsUmNGM+T2E6ubG3FvkY0HbLvNCcGRMCnDYqloo2BthuSG58KbMAku7VD
hMxzzJte5yMuRpHBojQr9N0hZcSJCtSFo6FhlxCzxGzwoSKzqm99ULYL4lf+IORe
DaKoATZJvbnX0LYAKLtPr1Pz9SD9pQIDAQABo4HUMIHRMAsGA1UdDwQEAwIHgDAd
BgNVHQ4EFgQUHpdxehvJwYL2o25SXyxsSndDW48wgY4GA1UdIwSBhjCBg4AUHpdx
ehvJwYL2o25SXyxsSndDW4+hYKReMFwxCzAJBgNVBAYTAlVTMRkwFwYDVQQIExBX
YXNoaW5ndG9uIFN0YXRlMRAwDgYDVQQHEwdTZWF0dGxlMSAwHgYDVQQKExdBbWF6
b24gV2ViIFNlcnZpY2VzIExMQ4IJAODRxRi5VXBbMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDQYJKoZIhvcNAQELBQADggEBAFGcArNlGB3oRBaXohpnhrnVblhJJa6U7r1/
M5TAJCKCkziSz3lOZmZZ1YgSnQC1blhUD2rhuS8a0Mw1Ml48nLjIYMMK0tkJRKCD
LPxxEZp4EzogYj6PqfSubzm59MmBenIcJPaJ9y2uY1U4Q3PMfU6M1JqqQ0Lzplq0
n9eWpJK3+7UbtuyAzigO4DRWEBMxJBUU7f2LfY+qFjskoeHopX20cmn1vc1TGyAF
Qf8xdt1SaPKC26zJsrCkiMZJNm4aa8KqvhqBHVjV4NrGeVGCcdlmSDiL71Egk9kh
744eGQxNnhZNvePW2kVcSuh0HpAo4Q4vfpjN+jz2Ves3kjFsrgc=
-----END CERTIFICATE-----
```

# Kepatuhan STIG untuk instans EC2 Anda

Security Technical Implementation Guides (STIGs) adalah standar pengerasan konfigurasi yang dibuat oleh Badan Sistem Informasi Pertahanan (DISA) untuk mengamankan sistem informasi dan perangkat lunak. Untuk membuat sistem Anda sesuai dengan standar STIG, Anda harus menginstal, mengonfigurasi, dan menguji berbagai pengaturan keamanan.

STIGs diterapkan berdasarkan klasifikasi kerentanan berikut. Saat Anda membuat gambar yang diperkeras STIG atau menerapkan pengaturan STIG ke instance yang ada, Anda dapat memilih level mana yang akan diterapkan untuk pengerasan. Level yang lebih tinggi termasuk pengaturan STIG dari semua level yang lebih rendah. Misalnya, level Tinggi (Kategori I) mencakup pengaturan dari kategori Sedang dan Rendah.

Tingkat kepatuhan

- Tinggi (Kategori I)

  Risiko paling parah. Termasuk kerentanan apa pun yang dapat mengakibatkan hilangnya kerahasiaan, ketersediaan, atau integritas.
- Sedang (Kategori II)

  Termasuk kerentanan apa pun yang dapat mengakibatkan hilangnya kerahasiaan, ketersediaan, atau integritas tetapi risikonya dapat dikurangi.
- Rendah (Kategori III)

  Termasuk kerentanan apa pun yang menurunkan tindakan untuk melindungi terhadap hilangnya kerahasiaan, ketersediaan, atau integritas.

Amazon EC2 menyediakan metode berikut untuk membuat instans yang dikeraskan STIG:

- Anda dapat meluncurkan instance Windows yang telah dikonfigurasi sebelumnya untuk pengerasan STIG dari Windows publik khusus. AWS AMIs Untuk informasi selengkapnya, lihat STIG Hardened AWS Windows Server AMIs di referensi AWS AMI Windows.
- Anda dapat membuat AMI yang disesuaikan untuk meluncurkan instans pra-konfigurasi yang dibuat dengan komponen pengerasan STIG EC2 Image Builder. Untuk informasi selengkapnya, lihat komponen pengerasan STIG di Panduan Pengguna Image Builder.
- Anda dapat menggunakan dokumen perintah AWSEC2-ConfigureSTIG Systems Manager untuk menerapkan pengaturan STIG ke instans EC2 yang ada. Dokumen perintah STIG Systems Manager memindai kesalahan konfigurasi dan menjalankan skrip remediasi untuk menginstal dan memperbarui sertifikat DoD dan menghapus sertifikat yang tidak perlu untuk mempertahankan kepatuhan STIG. Tidak ada biaya tambahan untuk menggunakan dokumen STIG Systems Manager.

Topik

- Pengaturan pengerasan STIG untuk instans EC2
- Unduhan skrip pengerasan STIG
- Gunakan AWS Systems Manager untuk menerapkan pengaturan STIG ke instans Anda


# Pengaturan pengerasan STIG untuk instans EC2

Pengaturan STIG bervariasi berdasarkan sistem operasi instans Anda dan tingkat kepatuhan yang Anda pilih. Konten berikut berisi detail spesifik sistem untuk STIGs yang diterapkan pada setiap tingkat kepatuhan.

Daftar Isi


- Pengaturan STIG untuk instans Linux EC2
- Pengaturan pengerasan STIG untuk instans Windows EC2


## Pengaturan STIG untuk instans Linux EC2

Bagian ini berisi informasi tentang pengaturan pengerasan STIG Linux yang didukung Amazon EC2. Skrip pengerasan menerapkan pengaturan STIG yang didukung ke infrastruktur berdasarkan distribusi Linux. Jika distribusi Linux tidak memiliki pengaturan pengerasan STIG sendiri, Amazon EC2 menggunakan pengaturan RHEL, sebagai berikut.

- Red Hat Enterprise Linux (RHEL) 7 pengaturan STIG
  - RHEL 7
  - CentOS 7
  - Amazon Linux 2 (AL2)
- Pengaturan RHEL 8 STIG
  - RHEL 8
  - CentOS 8
- Pengaturan RHEL 9 STIG
  - RHEL 9
  - CentOS Aliran 9

Linux STIG Rendah (Kategori III)

Daftar berikut berisi pengaturan STIG yang didukung Amazon EC2 ke infrastruktur Anda. Jika setelan yang didukung tidak berlaku untuk infrastruktur Anda, Amazon EC2 melewatkan setelan itu, dan melanjutkan. Misalnya, beberapa pengaturan pengerasan STIG mungkin tidak berlaku untuk server mandiri. Kebijakan khusus organisasi juga dapat memengaruhi pengaturan mana yang berlaku, seperti persyaratan bagi administrator untuk meninjau setelan dokumen.

Untuk daftar lengkap, lihat Perpustakaan STIGs Dokumen. Untuk informasi tentang cara melihat daftar lengkap, lihat Alat Melihat STIG.

RHEL 7 STIG Versi 3 Rilis 15

- RHEL 7/CentOS 7/ AL2

  V-204452, V-204576, dan V-204605

RHEL 8 STIG Versi 2 Rilis 6

- RHEL 8/CentOS 8

  V-230241, V-230269, V-230270, V-230281, V-230285, V-230346, V-230381, V-230395, V-230468, V-230469, V-230485, V-230486, V-230491, V-230494, V-230495, V-230494 6, V-230497, V-230498, V-230499, dan V-244527

RHEL 9 STIG Versi 2 Rilis 7

- RHEL 9/CentOS Aliran 9

  V-257782, V-257795, V-257796, V-257824, V-257880, V-257946, V-257947, V-258037, V-258067, V-258069, V-258076, V-258138, dan V-258173

RHEL 10 STIG Versi 1 Rilis 1

V-280995, V-281203, dan V-281292

Amazon Linux 2023 STIG Versi 1 Rilis 1

V-274141

SLES 12 STIG Versi 3 Rilis 4

V-217108, V-217113, V-217140, V-217198, V-217209, V-217211, V-217212, V-217213, V-217214, V-217215, V-217216, V-217236, V-217237, V-217238, V-217239, V-217282, dan V-255915

SLES 15 STIG Versi 2 Rilis 6

V-234811, V-234850, V-234868, V-234873, V-234905, V-234907, V-234908, V-234909, V-234933, V-234934, V-234935, V-234936, V-234955, V-234963, V-234967, dan V-255921

Ubuntu 18.04 STIG Versi 2 Rilis 15

V-219163, V-219164, V-219165, V-219172, V-219173, V-219174, V-219175, V-219178, V-219179, V-219180, V-219210, V-219301, V-219327, V-219332, dan V-219333

Ubuntu 20.04 STIG Versi 2 Rilis 4

V-238202, V-238203, V-238221, V-238222, V-238223, V-238224, V-238226, V-238234, V-238235, V-238237, V-238308, V-238323, V-238357, V-238362, dan V-238373

Ubuntu 22.04 STIG Versi 2 Rilis 7

V-260472, V-260476, V-260479, V-260480, V-260481, V-260520, V-260521, V-260549, V-260550, V-260551, V-260552, V-260581, dan V-260596

Ubuntu 24.04 STIG Versi 1 Rilis 4

V-270645, V-270646, V-270664, V-270677, V-270690, V-270695, V-270706, V-270710, V-270734, V-270749, V-270752, V-270818, dan V-270820

Linux STIG Medium (Kategori II)

Daftar berikut berisi pengaturan STIG yang didukung Amazon EC2 ke infrastruktur Anda. Jika setelan yang didukung tidak berlaku untuk infrastruktur Anda, Amazon EC2 melewatkan setelan itu, dan melanjutkan. Misalnya, beberapa pengaturan pengerasan STIG mungkin tidak berlaku untuk server mandiri. Kebijakan khusus organisasi juga dapat memengaruhi pengaturan mana yang berlaku, seperti persyaratan bagi administrator untuk meninjau setelan dokumen.

Untuk daftar lengkap, lihat Perpustakaan STIGs Dokumen. Untuk informasi tentang cara melihat daftar lengkap, lihat Alat Melihat STIG.

> **Note**
>
> Kategori Linux STIG Medium mencakup semua pengaturan pengerasan STIG terdaftar yang berlaku untuk Linux STIG Low (Kategori III), selain pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II.

RHEL 7 STIG Versi 3 Rilis 15

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

- RHEL 7/CentOS 7/ AL2

  V-204405, V-204406, V-204407, V-204408, V-204409, V-204410, V-204411, V-204412, V-204413, V-204414, V-204415, V-204416, V-204417, V-204420, V-204422, V-204423, V-204426, V-204427, V-204431, V-204434, V-204435, V-204437, V-204449, V-204450, V-204451, V-204457, V-204466, V-204490, V-204491, V-204503, V-204507, V-204508, V-204510, V-204511, V-204511, V-204511, V-204511 4512, V-204514, V-204515, V-204516, V-204517, V-204521, V-204524, V-204531, V-204536, V-204537, V-204538, V-204539, V-204540, V-204542, V-204543, V-204544, V-204545, V-204546, V-204546, V-204546, V-204546, V-204547, V-204548, V-204549, V-204550, V-204551, V-204552, V-204553, V-204554, V-204555, V-204556, V-204557, V-204558, V-204559, V-204560, V-204562, V-204563, V-204564, V-204566, V-204567, V-204568, V-204572, V-204579, V-204584, V-204585, V-204587, V-204588, V-204589, V-204590, V-204591, V-204592, V-204593, V-204596, V-204597, V-204598, V-204599, V-204601, V-204602, V-204609, V-204609, V-204610, V-204611, V-204612, V-204613, V-204614, V-204615, V-204616, V-204617, V-204619, V-204622, V-204625, V-204630, V-204631, V-204633, V-233307, V-237634, V-237635, V-251703, V-255925, V-255927, V-255928, dan V-256970

RHEL 8 STIG Versi 2 Rilis 6

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

- RHEL 8/CentOS 8

  V-230222, V-230228, V-230231, V-230233, V-230236, V-230237, V-230238, V-230239, V-230240, V-230240, V-230243, V-230244, V-230245, V-230246, V-230247, V-230248, V-230249, V-230250, V-230255, V-230256, V-230257, V-230258, V-230259, V-230260, V-230261, V-230262, V-230266,

V-230267, V-230268, V-230271, V-230273, V-230275, V-230275, V-230275 230276, V-230277, V-230278, V-230279, V-230280, V-230282, V-230282, V-230286, V-230287, V-230288, V-230290, V-230291, V-230296, V-230298, V-230310, V-230311, V-230312, V-230313, V-230314, V-230315, V-230316, V-230318, V-230319, V-230320, V-230321, V-230322, V-230324, V-230325, V-230326, V-230327, V-230330, V-230332, V-230333, V-230335, V-230337, V-230339, V-230341, V-230343, V-230345, V-230347, V-230347, V-230347 48, V-230352, V-230353, V-230354, V-230356, V-230357, V-230358, V-230359, V-230360, V-230361, V-230362, V-230363, V-230365, V-230366, V-230368, V-230369, V-230369, V-2303309 70, V-230372, V-230373, V-230375, V-230376, V-230377, V-230378, V-230380, V-230382, V-230383, V-230385, V-230386, V-230387, V-230389, V-230390, V-230392, V-230392, V-230392 30393, V-230394, V-230396, V-230397, V-230398, V-230399, V-230400, V-230401, V-230402, V-230403, V-230404, V-230405, V-230406, V-230407, V-230408, V-230409, V-230410, V-230411, V-230412, V-230413, V-230418, V-230419, V-230421, V-230422, V-230423, V-230423, V-230423, V-230423 24, V-230425, V-230426, V-230427, V-230428, V-230429, V-230430, V-230431, V-230432, V-230433, V-230434, V-230435, V-230436, V-230437, V-230438, V-230439, V-230439, V-230439 30444, V-230446, V-230447, V-230448, V-230449, V-230455, V-230456, V-230462, V-230463, V-230464, V-230465, V-230466, V-230467, V-230470, V-230471, V-230472, V-230473, V-230474, V-230475, V-230478, V-230480, V-230481, V-230482, V-230483, V-230488, V-230489, V-230493, V-230502, V-230503, V-230505, V-230506, V-230507, V-230523, V-230524, V-230525, V-230526, V-230527, V-230532, V-230535, V-230536, V-230537, V-230537, V-230537, V-230537, V-230537, V-230537 38, V-230539, V-230540, V-230541, V-230542, V-230543, V-230544, V-230545, V-230546, V-230547, V-230548, V-230549, V-230550, V-230555, V-230556, V-230557, V-230557, V-230557, V-230557 559, V-230560, V-230561, V-237640, V-237642, V-237643, V-244519, V-244523, V-244524, V-244525, V-244526, V-244528, V-244531, V-244533, V-244535, V-244536, V-244538, V-244539, V-244542, V-244543, V-244544, V-244545, V-244547, V-244550, V-244551, V-244552, V-244553, V-244554, V-250315, V-250315, V-250316, V-250316, V-250317, V-251707, V-251708, V-251709, V-251710, V-251711, V-251713, V-251714, V-251715, V-251716, V-251717, V-251718, V-256974, V-257258, V-257258, V-274877, V-279929, V-279930, dan V-279931

RHEL 9 STIG Versi 2 Rilis 7

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

- RHEL 9/CentOS Aliran 9

V-257780, V-257781, V-257783, V-257786, V-257788, V-257790, V-257791, V-257792, V-257793, V-257794, V-257797, V-257798, V-257799, V-257801, V-257802, V-257803, V-257804, V-257804, V-257805, V-257806, V-257807, V-257808, V-257809, V-257810, V-257811, V-257812, V-257813, V-257814, V-257815, V-257816, V-257817, V-257825, V-257827, V-257828, V-257829, V-257830, V-257831, V-257832, V-257833, V-257834, V-257836, V-257838, V-257839, V-257840, V-257841, V-257842, V-257849, V-257882, V-257883, V-257884, V-257885, V-257886, V-257886, V-257886 887, V-257888, V-257889, V-257890, V-257891, V-257892, V-257893, V-257894, V-257895, V-257896, V-257897, V-257898, V-257899, V-257900, V-257901, V-257902, V-257903, V-257904, V-257905, V-257906, V-257907, V-257907, V-257907, V-257907 7908, V-257909, V-257910, V-257911, V-257912, V-257913, V-257914, V-257915, V-257916, V-257917, V-257918, V-257919, V-257920, V-257921, V-257922, V-257923, V-257924, V-257925, V-257926, V-257927, V-257928, V-257929, V-257930, V-257933, V-257934, V-257935, V-257936, V-257939, V-257940, V-257942, V-257943, V-257944, V-257948, V-257951, V-257952, V-257953, V-257954, V-257954, V-257954, V-257954 57, V-257958, V-257959, V-257960, V-257961, V-257962, V-257963, V-257964, V-257965, V-257966, V-257967, V-257968, V-257969, V-257970, V-257971, V-257972, V-257973, V-257975, V-257976, V-257977, V-257978, V-257978, V-257979, V-257980, V-257982, V-257983, V-257985, V-257987, V-257988, V-257992, V-257993, V-257994, V-257995, V-257996, V-257997, V-257999, V-258000, V-258001, V-258001, V-258001, V-258001, V-258001 8002, V-258003, V-258004, V-258005, V-258006, V-258007, V-258008, V-258009, V-258010, V-258011, V-258028, V-258034, V-258035, V-258038, V-258039, V-258040, V-258041, V-258043, V-258046, V-258046, V-258046, V-258046, V-258046, V-258049, V-258052, V-258054, V-258055, V-258056, V-258057, V-258060, V-258063, V-258064, V-258065, V-258066, V-258068, V-258070, V-258071, V-258072, V-258073, V-258074, V-258075, V-258077, V-258079, V-258080, V-258081, V-258082, V-258083, V-258084, V-258085, V-258088, V-258089, V-258090, V-258091, V-258092, V-258093, V-258095, V-258097, V-258098, V-258099, V-258100, V-258101, V-258102, V-258102, V-258102 103, V-258104, V-258105, V-258107, V-258108, V-258109, V-258110, V-258111, V-258112, V-258113, V-258114, V-258115, V-258116, V-258117, V-258118, V-258119, V-258120, V-258120, V-258120, V-258120, V-258120, V-258120 8121, V-258122, V-258123, V-258124, V-258125, V-258126, V-258128, V-258129, V-258130, V-258133, V-258137, V-258140, V-258141, V-258142, V-258144, V-258146, V-258147, V-258148, V-258148, V-258148 150, V-258151, V-258152, V-258153, V-258154, V-258156, V-258157, V-258158, V-258159, V-258160, V-258161, V-258162, V-258163, V-258164, V-258165, V-258166, V-258166, V-258166 167, V-258168, V-258169, V-258170, V-258171, V-258172, V-258175, V-258176, V-258177, V-258178, V-258179, V-258180, V-258181, V-258182, V-258183, V-258184, V-258184, V-258184 5, V-258186, V-258187, V-258188, V-258189, V-258190, V-258191, V-258192, V-258193, V-258194, V-258195, V-258196, V-258197, V-258198, V-258199, V-258200, V-258201, V-258202, V-258203, V-258204, V-258206,

V-258207, V-258208, V-258209, V-258210, V-258211, V-258212, V-258213, V-258214, V-258215, V-258216, V-258217, V-258218, V-258219, V-258220, V-258221, V-258222, V-258223, V-258224, V-258225, V-258226, V-258227, V-258227, V-258227 28, V-258229, V-258232, V-258233, V-258234, V-258237, V-258239, V-258240, V-270174, V-270175, V-270176, V-270177, V-272488, dan V-279936

RHEL 10 STIG Versi 1 Rilis 1

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-280094, V-280942, V-280943, V-280945, V-280946, V-280947, V-280948, V-280950, V-280952, V-280953, V-280954, V-280955, V-280956, V-280958, V-280959, V-280960, V-280961, V-280962, V-280963, V-280964, V-280965, V-280966, V-280967, V-280968, V-280969, V-280970, V-280972, V-280973, V-280974, V-280975, V-280976, V-280976, V-280976 977, V-280981, V-280983, V-280984, V-280986, V-280987, V-280988, V-280989, V-280990, V-280991, V-280992, V-280993, V-280994, V-280996, V-280997, V-280999, V-280999 281000, V-281001, V-281002, V-281003, V-281005, V-281006, V-281017, V-281018, V-281019, V-281020, V-281021, V-281022, V-281023, V-281024, V-281025, V-281026, V-281027, V-281028, V-281029, V-281030, V-281031, V-281032, V-281033, V-281034, V-281035, V-281036, V-281037, V-281038, V-281039, V-281040, V-281041, V-281042, V-281043, V-281044, V-281045, V-281047, V-281048, V-281049, V-281050, V-281051, V-281052, V-281053, V-281054, V-281055, V-281056, V-281057, V-281058, V-281058, V-281058, V-281058 59, V-281060, V-281061, V-281062, V-281063, V-281064, V-281065, V-281066, V-281067, V-281068, V-281069, V-281070, V-281071, V-281072, V-281073, V-281074, V-281075, V-281076, V-281077, V-281077, V-281077, V-281077, V-281077, V-281077 1078, V-281079, V-281080, V-281081, V-281082, V-281083, V-281084, V-281085, V-281086, V-281087, V-281094, V-281095, V-281096, V-281097, V-281098, V-281099, V-281100, V-281101, V-281102, V-281102, V-281102 3, V-281104, V-281105, V-281107, V-281108, V-281109, V-281110, V-281111, V-281113, V-281114, V-281115, V-281116, V-281117, V-281118, V-281119, V-281120, V-281121, V-281122, V-281123, V-281124, V-281125, V-281126, V-281127, V-281128, V-281129, V-281130, V-281131, V-281132, V-281133, V-281134, V-281135, V-281136, V-281137, V-281138, V-281139, V-281139, V-281139, V-281139, 281140, V-281141, V-281142, V-281143, V-281144, V-281145, V-281146, V-281147, V-281148, V-281149, V-281150, V-281151, V-281152, V-281153, V-281154, V-281155, V-281156, V-281157, V-281158, V-281159, V-281160, V-281161, V-281162, V-281163, V-281164, V-281165, V-281168, V-281169, V-281171, V-281174, V-281175, V-281177, V-281178, V-281179, V-281180, V-281181, V-281181, V-281181, 281182, V-281183, V-281184, V-281185, V-281187, V-281188, V-281189, V-281190, V-281191, V-281192, V-281193, V-281195, V-281196, V-281197,

V-281198, V-281200, V-281200, V-281198 1201, V-281204, V-281205, V-281206, V-281209, V-281210, V-281212, V-281214, V-281215, V-281217, V-281218, V-281219, V-281220, V-281224, V-281225, V-281226, V-281249, V-281251, V-281252, V-281253, V-281254, V-281255, V-281256, V-281257, V-281258, V-281259, V-281260, V-281261, V-281262, V-281263, V-281264, V-281265, V-281266, V-281269, V-281270, V-281271, V-281272, V-281273, V-281274, V-281276, V-281277, V-281278, V-281279, V-281280, V-281281, V-281282, V-281282, V-281282, V-281282 V-281285, V-281286, V-281287, V-281288, V-281289, V-281290, V-281291, V-281293, V-281295, V-281296, V-281297, V-281300, V-281301, V-281302, V-281303, V-281304, V-281305, V-281306, V-281306, V-281306, V-281306, V-281306, V-281306 1308, V-281309, V-281310, V-281311, V-281312, V-281313, V-281314, V-281315, V-281316, V-281317, V-281318, V-281319, V-281320, V-281322, V-281324, V-281325, V-281327, V-281328, V-281331, V-281333, V-281334, V-281335, V-281337, V-281338, V-281340, V-281341, V-281342, V-281343, V-281344, V-281345, V-281346, V-281348, V-281349, V-281350, V-281351, V-281352, V-281352, V-281352 53, V-281354, V-281355, V-281356, V-281357, V-281358, V-281359, V-281360, V-281361, V-281363, V-281364, dan V-281365

Amazon Linux 2023 STIG Versi 1 Rilis 2

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-273995, V-274000, V-274001, V-274002, V-274003, V-274004, V-274005, V-274006, V-274008, V-274009, V-274010, V-274011, V-274012, V-274013, V-274014, V-274017, V-274018, V-274019, V-274020, V-274020, V-274021, V-274022, V-274023, V-274024, V-274026, V-274027, V-274028, V-274030, V-274031, V-274032, V-274033, V-274034, V-274035, V-274036, V-274040, V-274041, V-274042, V-274044, V-274045, V-274047, V-274048, V-274049, V-274050, V-274051, V-274053, V-274054, V-274059, V-274061, V-274062, V-274069, V-274070, V-274071, V-274072, V-274073, V-274074, V-274075, V-274075, V-274075 76, V-274077, V-274078, V-274079, V-274081, V-274082, V-274083, V-274084, V-274085, V-274086, V-274087, V-274088, V-274089, V-274090, V-274091, V-274092, V-274093, V-274094, V-274095, V-274096, V-274097, V-274098, V-274099, V-274100, V-274101, V-274102, V-274103, V-274104, V-274105, V-274106, V-274107, V-274108, V-274109, V-274110, V-274111, V-274112, V-274113, V-274113, V-274113, V-274113, V-274113, V-274113, V-274113, V-274113 114, V-274115, V-274116, V-274117, V-274119, V-274120, V-274121, V-274122, V-274123, V-274124, V-274125, V-274126, V-274127, V-274129, V-274130, V-274131, V-274131, V-274131, V-274132, V-274133, V-274134, V-274135, V-274136, V-274137, V-274138, V-274139, V-274140, V-274143, V-274144, V-274145, V-274147, V-274149, V-274151, V-274152, V-274154, V-274155, V-274156, V-274157, V-274160, V-274161, V-274162,

V-274163, V-274164, V-274165, V-274166, V-274167, V-274168, V-274169, V-274170, V-274173, V-274177, V-274181, V-274182, V-274185, dan V-274187

SLES 12 STIG Versi 3 Rilis 4

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-217102, V-217105, V-217105, V-217106, V-217106, V-217110, V-217116, V-217117, V-217118, V-217119, V-217120, V-217121, V-217122, V-217124, V-217125, V-217126, V-217127, V-217128, V-217130, V-217134, V-217138, V-217143, V-217147, V-217152, V-217153, V-217154, V-217155, V-217156, V-217158, V-217161, V-217163, V-217166, V-217167, V-217168, V-217169, V-217170, V-217171, V-217182, V-217183, V-217188, V-217190, V-217191, V-217194, V-217195, V-217196, V-217197, V-217200, V-217201, V-217202, V-217202, V-217202, 217203, V-217204, V-217205, V-217206, V-217207, V-217208, V-217210, V-217217, V-217218, V-217223, V-217227, V-217230, V-217240, V-217241, V-217242, V-217243, V-217244, V-217245, V-217246, V-217247, V-217248, V-217249, V-217250, V-217251, V-217251, V-217251, 217252, V-217253, V-217254, V-217255, V-217257, V-217258, V-217260, V-217265, V-217266, V-217267, V-217269, V-217272, V-217273, V-217274, V-217275, V-217276, V-217277, V-217278, V-217279, V-217280, V-217283, V-217284, V-217286, V-217287, V-217288, V-217289, V-217290, V-217291, V-217292, V-217293, V-217294, V-217295, V-217296, V-217299, V-217300, V-217301, V-217302, V-233308, V-237605, V-237606, V-237607, V-237608, V-237609, V-237610, V-237611, V-237612, V-237614, V-237615, V-237616, V-237617, V-237618, V-237619, V-237620, V-237621, V-237622, V-237623, V-251720, V-251722, V-251722, V-251722, V-251722 5914, dan V-256981

SLES 15 STIG Versi 2 Rilis 6

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-234802, V-234807, V-234808, V-234809, V-234813, V-234815, V-234817, V-234821, V-234822, V-234823, V-234825, V-234827, V-234828, V-234829, V-234830, V-234832, V-234833, V-234834, V-234835, V-234836, V-234837, V-234838, V-234839, V-234840, V-234841, V-234842, V-234843, V-234844, V-234845, V-234848, V-234854, V-234855, V-234856, V-234857, V-234858, V-234861, V-234861, V-234862, V-234863, V-234869, V-234870, V-234875, V-234878, V-234880, V-234881, V-234882, V-234883, V-234884, V-234885, V-234886, V-234887, V-234888, V-234889, V-234891, V-234895, V-234896, V-234896, V-234896, V-234896 97, V-234899, V-234900, V-234901, V-234902, V-234903, V-234904, V-234906, V-234910, V-234911, V-234912, V-234913, V-234914, V-234918, V-234924, V-234928, V-234932, V-234937, V-234938, V-234939, V-234940, V-234941,

V-234942, V-234943, V-234944, V-234945, V-234946, V-234947, V-234948, V-234949, V-234950, V-234951, V-234952, V-234954, V-234956, V-234957, V-234958, V-234959, V-234959, V-234959, V-234959, V-234959, V-234959 61, V-234962, V-234964, V-234966, V-234969, V-234973, V-234975, V-234976, V-234977, V-234978, V-234979, V-234981, V-234982, V-234983, V-234991, V-235002, V-235003, V-235007, V-235008, V-235008, V-235008, V-235009, V-235010, V-235013, V-235014, V-235015, V-235016, V-235017, V-235018, V-235019, V-235020, V-235021, V-235022, V-235023, V-235024, V-235025, V-235026, V-235028, V-235029, V-235030, V-251724, V-255920, V-256983, dan V-274879

Ubuntu 18.04 STIG Versi 2 Rilis 15

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-219149, V-219155, V-219156, V-219160, V-219166, V-219168, V-219176, V-219181, V-219184, V-219186, V-219188, V-219189, V-219190, V-219191, V-219192, V-219193, V-219194, V-219195, V-219196, V-219197, V-219198, V-219199, V-219200, V-219201, V-219202, V-219203, V-219204, V-219205, V-219206, V-219207, V-219208, V-219209, V-219213, V-219214, V-219215, V-219216, V-219217, V-219218, V-219219, V-219220, V-219221, V-219222, V-219223, V-219224, V-219225, V-219226, V-219227, V-219227, V-219227, V-219227, V-219227, V-219227, V-219227, V-219227 28, V-219229, V-219230, V-219231, V-219232, V-219233, V-219234, V-219235, V-219236, V-219238, V-219239, V-219240, V-219241, V-219242, V-219243, V-219244, V-219250, V-219254, V-219257, V-219263, V-219264, V-219265, V-219266, V-219267, V-219268, V-219269, V-219270, V-219271, V-219272, V-219273, V-219274, V-219275, V-219276, V-219277, V-219279, V-219281, V-219287, V-219291, V-219296, V-219297, V-219298, V-2192198 99, V-219300, V-219303, V-219304, V-219306, V-219309, V-219310, V-219311, V-219315, V-219318, V-219319, V-219323, V-219326, V-219328, V-219330, V-219331, V-2319331, V-219331, V-219331, V-219331, V-219331 35, V-219336, V-219337, V-219338, V-219339, V-219342, V-219344, V-233779, V-233780, dan V-255906

Ubuntu 20.04 STIG Versi 2 Rilis 4

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-238200, V-238205, V-238207, V-238209, V-238210, V-238211, V-238212, V-238213, V-238220, V-238225, V-238227, V-238228, V-238229, V-238230, V-238231, V-238232, V-238236, V-238238, V-238239, V-238240, V-238241, V-238242, V-238244, V-238245, V-238246, V-238247, V-238248, V-238249, V-238250, V-238251, V-238252, V-238253, V-238254, V-238255, V-238256, V-238257,

V-238258, V-238264, V-238268, V-238271, V-238277, V-238278, V-238279, V-238280, V-238281, V-238282, V-238283, V-238284, V-238285, V-238286, V-238287, V-238288, V-238289, V-238290, V-238291, V-238292, V-238293, V-238294, V-238295, V-238297, V-238298, V-238300, V-238301, V-238302, V-238303, V-238303, V-238309, V-238310, V-238315, V-238316, V-238317, V-238318, V-238318, V-238318 238319, V-238320, V-238324, V-238325, V-238329, V-238333, V-238334, V-238337, V-238338, V-238339, V-238340, V-238341, V-238342, V-238343, V-238344, V-238345, V-238346, V-238347, V-238348, V-238349, V-238350, V-238351, V-238352, V-238353, V-238355, V-238356, V-238359, V-238369, V-238370, V-238371, V-238376, V-238377, V-238378, V-251505, V-255912, V-274852, dan V-274853

Ubuntu 22.04 STIG Versi 2 Rilis 7

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-260471, V-260473, V-260474, V-260475, V-260475, V-260477, V-260478, V-260485, V-260486, V-260487, V-260488, V-260489, V-260490, V-260491, V-260492, V-260492, V-260492, V-260492, V-260492, V-260492 3, V-260494, V-260495, V-260496, V-260497, V-260498, V-260499, V-260500, V-260505, V-260506, V-260507, V-260508, V-260509, V-260510, V-260511, V-260512, V-260512, V-260512, V-260512 513, V-260514, V-260522, V-260527, V-260528, V-260530, V-260533, V-260534, V-260535, V-260537, V-260538, V-260540, V-260542, V-260543, V-260545, V-260546, V-260546, V-260546, V-260546, V-260546, V-260547, V-260553, V-260554, V-260555, V-260556, V-260557, V-260560, V-260561, V-260562, V-260562, V-260563, V-260564, V-260565, V-260566, V-260567, V-260569, V-260572, V-260573, V-260574, V-260575, V-260576, V-260582, V-260584, V-260586, V-260588, V-260588, V-260588, V-260588, V-260588, V-260588 89, V-260590, V-260591, V-260594, V-260597, V-260598, V-260599, V-260600, V-260601, V-260602, V-260603, V-260604, V-260605, V-260606, V-260607, V-260608, V-260608, V-260608, V-260608 609, V-260610, V-260611, V-260612, V-260613, V-260614, V-260615, V-260616, V-260617, V-260618, V-260619, V-260620, V-260621, V-260622, V-260623, V-260624, V-260624, V-260624, V-260624, V-260624, V-260625, V-260626, V-260627, V-260628, V-260629, V-260630, V-260631, V-260632, V-260633, V-260634, V-260635, V-260636, V-260637, V-260638, V-260639, V-260640, V-260641, V-260642, V-260643, V-260644, V-260646, V-260647, V-260648, V-260649, V-274862, V-274864, dan V-274866

Ubuntu 24.04 STIG Versi 1 Rilis 4

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-270649, V-270651, V-270652, V-270653, V-270654, V-270656, V-270657, V-270659, V-270660, V-270661, V-270662, V-270663, V-270669, V-270672, V-270673, V-270674, V-270676, V-270678, V-270679, V-270680, V-270681, V-270683, V-270684, V-270685, V-270686, V-270687, V-270688, V-270689, V-270692, V-270693, V-270696, V-270697, V-270698, V-270700, V-270701, V-270702, V-270702, V-270702, V-270702, V-270702, V-270703, V-270704, V-270705, V-270709, V-270715, V-270716, V-270718, V-270720, V-270721, V-270722, V-270723, V-270724, V-270725, V-270726, V-270727, V-270728, V-270729, V-270730, V-270730, V-270730, V-270730, V-270730, V-270731, V-270732, V-270733, V-270737, V-270739, V-270740, V-270741, V-270742, V-270743, V-270746, V-270750, V-270753, V-270755, V-270756, V-270757, V-270758, V-270759, V-270760, V-270765, V-270766, V-270767, V-270768, V-270769, V-270770, V-270771, V-270772, V-270773, V-270775, V-270776, V-270777, V-270778, V-270779, V-270780, V-270781, V-270782, V-270783, V-270784, V-270785, V-270786, V-270786, V-270786 87, V-270788, V-270789, V-270790, V-270791, V-270792, V-270793, V-270794, V-270795, V-270796, V-270797, V-270798, V-270799, V-270800, V-270801, V-270802, V-270803, V-270804, V-270804, V-270804, V-270804 805, V-270806, V-270807, V-270808, V-270809, V-270810, V-270811, V-270813, V-270814, V-270815, V-270821, V-270822, V-270823, V-270824, V-270825, V-270826, V-270827, V-270828, V-270828, V-270829, V-270829, 270830, V-270831, V-270832, V-274870, V-274871, V-274872, dan V-274873

Linux STIG Tinggi (Kategori I)

Daftar berikut berisi pengaturan STIG yang didukung Amazon EC2 ke infrastruktur Anda. Jika setelan yang didukung tidak berlaku untuk infrastruktur Anda, Amazon EC2 melewatkan setelan itu, dan melanjutkan. Misalnya, beberapa pengaturan pengerasan STIG mungkin tidak berlaku untuk server mandiri. Kebijakan khusus organisasi juga dapat memengaruhi pengaturan mana yang berlaku, seperti persyaratan bagi administrator untuk meninjau setelan dokumen.

Untuk daftar lengkap, lihat Perpustakaan STIGs Dokumen. Untuk informasi tentang cara melihat daftar lengkap, lihat Alat Melihat STIG.

> **Note**
>
> Kategori Linux STIG High mencakup semua pengaturan pengerasan STIG terdaftar yang berlaku untuk kategori Linux STIG Medium dan Low, selain pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori I.

RHEL 7 STIG Versi 3 Rilis 15

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

- RHEL 7/CentOS 7/ AL2

  V-204424, V-204425, V-204442, V-204443, V-204447, V-204448, V-204455, V-204462, V-204497, V-204497, V-204502, V-204594, V-204620, dan V-204621

RHEL 8 STIG Versi 2 Rilis 6

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

- RHEL 8/CentOS 8

  V-230223, V-230264, V-230283, V-230284, V-230487, V-230492, V-230533, V-230558, V-244540, V-279933, V-230265, V-230226, V-230530, V-268322, V-230529, dan V-230531

RHEL 9 STIG Versi 2 Rilis 7

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

- RHEL 9/CentOS Aliran 9

  V-257820, V-257821, V-257826, V-257835, V-257955, V-257956, V-258059, V-258230, V-258238, V-257984, V-257986, V-258078, V-258094, V-258235, V-257784, dan V-257785

RHEL 10 Versi 1 Rilis 1

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-280932, V-280933, V-280934, V-280944, V-280949, V-280951, V-280978, V-281007, V-281014, V-281015, V-281216, V-281222, V-281223, V-281267, V-281268, V-281275, V-281298, V-281298, V-281298 1299, dan V-281307

Amazon Linux 2023 STIG Versi 1 Rilis 2

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-273996, V-273997, V-273999, V-274007, V-274038, V-274039, V-274046, V-274052, V-274057, dan V-274153

SLES 12 STIG Versi 3 Rilis 4

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-217101, V-217139, V-217141, V-217142, V-217159, V-217160, V-217164, V-217264, V-217268, V-222386, dan V-251721

SLES 15 STIG Versi 2 Rilis 6

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-234800, V-234804, V-234818, V-234852, V-234859, V-234860, V-234898, V-234984, V-234985, V-234988, V-234989, V-234990, V-235031, V-235032, dan V-251725

Ubuntu 18.04 STIG Versi 2 Rilis 15

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-219157, V-219158, V-219177, V-219212, V-219308, V-219314, V-219316, dan V-251507

Ubuntu 20.04 STIG Versi 2 Rilis 4

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-238201, V-238218, V-238219, V-238326, V-238327, V-238380, dan V-251504

Ubuntu 22.04 STIG Versi 2 Rilis 7

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-260469, V-260482, V-260483, V-260523, V-260524, V-260526, V-260529, V-260539, V-260570, V-260571, V-260579, dan V-279937

Ubuntu 24.04 STIG Versi 1 Rilis 4

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-270647, V-270648, V-270665, V-270666, V-270708, V-270711, V-270712, V-270713, V-270714, V-270717, V-270736, V-270738, dan V-279938

## Pengaturan pengerasan STIG untuk instans Windows EC2

Amazon EC2 mendukung garis dasar STIG untuk versi sistem operasi Windows berikut:

- Windows Server 2025
- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2

Bagian ini mencantumkan pengaturan STIG saat ini yang didukung Amazon EC2 untuk infrastruktur Windows Anda, diikuti oleh log riwayat versi.

Anda dapat menerapkan pengaturan STIG rendah, sedang, atau tinggi.

Windows STIG Rendah (Kategori III)

Daftar berikut berisi pengaturan STIG yang didukung Amazon EC2 ke infrastruktur Anda. Jika setelan yang didukung tidak berlaku untuk infrastruktur Anda, Amazon EC2 melewatkan setelan itu, dan melanjutkan. Misalnya, beberapa pengaturan pengerasan STIG mungkin tidak berlaku untuk server mandiri. Kebijakan khusus organisasi juga dapat memengaruhi pengaturan mana yang berlaku, seperti persyaratan bagi administrator untuk meninjau setelan dokumen.

Untuk daftar lengkap Windows STIGs, lihat Perpustakaan STIGs Dokumen. Untuk informasi tentang cara melihat daftar lengkap, lihat Alat Melihat STIG.

- Windows Server 2025 STIG Versi 1 Rilis 1

V-278082, V-278083, V-278084, V-278085, V-278098, V-278104, V-278110, dan V-278231

- Windows Server 2022 STIG Versi 2 Rilis 7

  V-254335, V-254336, V-254337, V-254338, V-254351, V-254357, V-254363, dan V-254481

- Windows Server 2019 STIG Versi 3 Rilis 7

  V-205691, V-205819, V-205858, V-205859, V-205860, V-205870, V-205871, dan V-205923

- Windows Server 2016 STIG Versi 2 Rilis 10

  V-224916, V-224917, V-224918, V-224919, V-224931, V-224942, dan V-225060

- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5

  V-225250, V-225318, V-225319, V-225324, V-225327, V-225330, V-225331, V-225332, V-225333, V-225334, V-225335, V-225336, V-225342, V-225343, V-225355, V-225357, V-225358, V-225359, V-225360, V-225362, V-225363, V-225376, V-225392, V-225394, V-225412, V-225460, V-225462, V-225468, V-225473, V-225473, V-225473 76, V-225479, V-225480, V-225481, V-225482, V-225483, V-225484, V-225485, V-225487, V-225488, V-225489, V-225490, V-225511, V-225514, V-225525, V-225526, V-225526, V-225526, V-225526 25536, dan V-225537

- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 7

  Tidak ada pengaturan STIG yang berlaku untuk kerentanan Microsoft .NET Framework untuk Kategori III.

- Windows Firewall STIG Versi 2 Rilis 2

  V-241994, V-241995, V-241996, V-241999, V-242000, V-242001, V-242006, V-242007, dan V-242008

- Internet Explorer 11 STIG Versi 2 Rilis 6

  V-223056 dan V-223078

- Microsoft Edge STIG Versi 2 Rilis 4 (hanya Windows Server 2022 dan 2025)

  V-235727, V-235731, V-235751, V-235752, dan V-235765

Windows STIG Medium (Kategori II)

Daftar berikut berisi pengaturan STIG yang didukung Amazon EC2 ke infrastruktur Anda. Jika setelan yang didukung tidak berlaku untuk infrastruktur Anda, Amazon EC2 melewatkan setelan itu, dan

melanjutkan. Misalnya, beberapa pengaturan pengerasan STIG mungkin tidak berlaku untuk server mandiri. Kebijakan khusus organisasi juga dapat memengaruhi pengaturan mana yang berlaku, seperti persyaratan bagi administrator untuk meninjau setelan dokumen.

Untuk daftar lengkap Windows STIGs, lihat Perpustakaan STIGs Dokumen. Untuk informasi tentang cara melihat daftar lengkap, lihat Alat Melihat STIG.

> **Note**
>
> Kategori Windows STIG Medium mencakup semua pengaturan pengerasan STIG terdaftar yang berlaku untuk Windows STIG rendah (Kategori III), selain pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II.

- Windows Server 2025 STIG Versi 1 Rilis 1

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

  V-278015, V-278016, V-278019, V-278020, V-278021, V-278022, V-278023, V-278024, V-278025, V-278026, V-278033, V-278034, V-278035, V-278036, V-278037, V-278038, V-278039, V-278047, V-278048, V-278049, V-278050, V-278051, V-278052, V-278053, V-278054, V-278055, V-278056, V-278057, V-278058, V-278059, V-278060, V-278061, V-278062, V-278063, V-278064, V-278065, V-278066, V-278066, V-278066, V-278066, V-278066 67, V-278068, V-278069, V-278070, V-278071, V-278072, V-278073, V-278074, V-278075, V-278076, V-278077, V-278078, V-278079, V-278080, V-278086, V-278089, V-278091, V-278092, V-278092, V-278092, V-278092, V-278092 8093, V-278094, V-278095, V-278096, V-278097, V-278102, V-278103, V-278105, V-278106, V-278107, V-278108, V-278109, V-278111, V-278112, V-278113, V-278114, V-278115, V-278116, V-278116, V-278116, V-278116 8117, V-278118, V-278119, V-278120, V-278122, V-278123, V-278124, V-278126, V-278127, V-278129, V-278130, V-278131, V-278165, V-278168, V-278169, V-278170, V-278178 1, V-278174, V-278180, V-278181, V-278182, V-278183, V-278184, V-278185, V-278187, V-278188, V-278189, V-278192, V-278193, V-278194, V-278195, V-278198, V-278198, V-278278 199, V-278200, V-278201, V-278202, V-278203, V-278204, V-278205, V-278206, V-278209, V-278210, V-278211, V-278212, V-278213, V-278214, V-278218, V-278220, V-278221, V-278222, V-278222, V-278223, V-278226, V-278228, V-278229, V-278230, V-278232, V-278233, V-278233, V-278233 8234, V-278235, V-278236, V-278237, V-278238, V-278239, V-278240, V-278241, V-278243, V-278244, V-278245, V-278247, V-278249, V-278251, V-278252, V-278252, V-278252 3, V-278254, V-278255, V-278256, V-278257, V-278258, V-278259,

V-278260, V-278261, V-278262, V-279916, V-279917, V-279918, V-279919, V-279920, V-279921, V-279922, dan V-279923

- Windows Server 2022 STIG Versi 2 Rilis 7

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

  V-254247, V-254269, V-254270, V-254271, V-254272, V-254273, V-254274, V-254275, V-254276, V-254277, V-254278, V-254285, V-254286, V-254287, V-254288, V-254288, V-254288, V-254288 9, V-254290, V-254291, V-254292, V-254296, V-254297, V-254298, V-254299, V-254300, V-254301, V-254302, V-254303, V-254305, V-254307, V-254309, V-254311, V-254312, V-254313, V-254314, V-254315, V-254316, V-254319, V-254320, V-254321, V-254322, V-254323, V-254324, V-254325, V-254326, V-254327, V-254328, V-254329, V-254330, V-254331, V-254331, V-254331, V-254331, V-254331 32, V-254333, V-254334, V-254339, V-254341, V-254342, V-254344, V-254345, V-254346, V-254347, V-254348, V-254349, V-254350, V-254355, V-254356, V-254358, V-254359, V-254360, V-254361, V-254362, V-254364, V-254365, V-254366, V-254367, V-254367, V-254367, V-254367, V-254367, V-254367 4368, V-254369, V-254370, V-254371, V-254372, V-254373, V-254375, V-254376, V-254377, V-254379, V-254380, V-254382, V-254383, V-254431, V-254433, V-254434, V-254434, V-254435, V-254436, V-254438, V-254439, V-254440, V-254442, V-254443, V-254444, V-254445, V-254447, V-254448, V-254449, V-254450, V-254451, V-254453, V-254454, V-254455, V-254456, V-254459, V-254460, V-254461, V-254462, V-254463, V-254464, V-254468, V-254470, V-254471, V-254472, V-254473, V-254476, V-254477, V-254478, V-254479, V-254480, V-254482, V-254483, V-254485, V-254486, V-254486, V-254486 V-254488, V-254489, V-254491, V-254493, V-254494, V-254495, V-254497, V-254498, V-254499, V-254501, V-254502, V-254503, V-254504, V-254505, V-254506, V-254506 7, V-254508, V-254509, V-254510, V-254511, V-254512, V-278942, V-278943, V-278944, V-278945, V-278946, V-278947, V-278948, dan V-278949

- Windows Server 2019 STIG Versi 3 Rilis 7

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

  V-205625, V-205626, V-205627, V-205629, V-205630, V-205633, V-205634, V-205635, V-205636, V-205637, V-205638, V-205639, V-205640, V-205641, V-205643, V-205644, V-205648, V-205649, V-205650, V-205651, V-205652, V-205655, V-205656, V-205659, V-205660, V-205662, V-205671, V-205672, V-205673, V-205675, V-205676, V-205678, V-205680, V-205681, V-205682, V-205682, V-205682, V-205682, V-205682 V-205684, V-205685, V-205686, V-205687, V-205688, V-205689,

V-205690, V-205692, V-205693, V-205694, V-205697, V-205698, V-205708, V-205709, V-205712, V-205714, V-205716, V-205717, V-205717, V-205718, V-205719, V-205720, V-205722, V-205730, V-205731, V-205733, V-205747, V-205748, V-205749, V-205751, V-205752, V-205754, V-205755, V-205756, V-205759, V-205760, V-205761, V-205761, V-205762, V-205763, V-205764, V-205765, V-205766, V-205767, V-205768, V-205769, V-205770, V-205771, V-205772, V-205773, V-205774, V-205775, V-205776, V-205778, V-205779, V-205780, V-205780, V-205780, V-205780, V-205780, V-205780 V-205782, V-205783, V-205784, V-205795, V-205796, V-205797, V-205798, V-205801, V-205808, V-205809, V-205810, V-205811, V-205812, V-205813, V-205814, V-205815, V-205816, V-205817, V-205821, V-205822, V-205823, V-205824, V-205825, V-205826, V-205827, V-205830, V-205832, V-205833, V-205833, V-205835, V-205836, V-205837, V-205842, V-205861, V-205863, V-205865, V-205866, V-205866, V-205867, V-205868, V-205869, V-205872, V-205873, V-205874, V-205909, V-205910, V-205911, V-205912, V-205915, V-205916, V-205918, V-205920, V-205921, V-205922, V-205925, V-2525 7503, V-278934, V-278935, V-278936, V-278937, V-278938, V-278939, V-278940, dan V-278941

- Windows Server 2016 STIG Versi 2 Rilis 10

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

  V-224850, V-224851, V-224852, V-224853, V-224854, V-224855, V-224856, V-224857, V-224858, V-224859, V-224866, V-224867, V-224869, V-224870, V-224870, V-224870, V-224870, V-224870 71, V-224872, V-224873, V-224877, V-224878, V-224879, V-224880, V-224881, V-224882, V-224883, V-224884, V-224885, V-224886, V-224888, V-224890, V-224892, V-224892, V-224892, V-224892 24893, V-224894, V-224895, V-224896, V-224897, V-224900, V-224901, V-224902, V-224903, V-224904, V-224905, V-224906, V-224907, V-224908, V-224909, V-224910, V-224910, V-224910 224911, V-224912, V-224913, V-224914, V-224915, V-224920, V-224921, V-224922, V-224924, V-224925, V-224926, V-224927, V-224928, V-224929, V-224930, V-224935, V-224936, V-224937, V-224938, V-224939, V-224940, V-224941, V-224943, V-224944, V-224945, V-224945, V-224945, V-224945, V-224945 46, V-224947, V-224948, V-224949, V-224951, V-224952, V-224953, V-224955, V-224956, V-224957, V-224959, V-224960, V-224962, V-224963, V-224965, V-224966, V-224966, V-224966, V-224966, V-224966 24967, V-224968, V-224969, V-224987, V-224988, V-224989, V-224995, V-224996, V-224997, V-224998, V-224999, V-225000, V-225001, V-225002, V-225003, V-225004, V-225004, V-225004, V-225004 25005, V-225008, V-225009, V-225010, V-225011, V-225013, V-225014, V-225015, V-225016, V-225017, V-225018, V-225019, V-225020, V-225021, V-225022, V-225023, V-225024, V-225026, V-225027, V-225028, V-225029, V-225030, V-225031, V-225032, V-225033, V-225034, V-225035, V-225038, V-225039, V-225040, V-225041, V-225042, V-225043, V-225047, V-225049, V-225050, V-225051, V-225052, V-225055,

V-225056, V-225057, V-225058, V-225059, V-225061, V-225062, V-225063, V-225064, V-225065, V-225066, V-225067, V-225068, V-225070, V-225072, V-225073, V-225074, V-225076, V-225077, V-225078, V-225080, V-225081, V-225082, V-225083, V-225083, V-225083 4, V-225085, V-225086, V-225087, V-225088, V-225089, V-225092, V-225093, dan V-257502

- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

  V-225239, V-225259, V-225260, V-225261, V-225263, V-225264, V-225265, V-225266, V-225267, V-225268, V-225269, V-225270, V-225271, V-225272, V-225273, V-225275, V-225276, V-225277, V-225278, V-225279, V-225280, V-225281, V-225282, V-225283, V-225284, V-225285, V-225286, V-225287, V-225288, V-225289, V-225290, V-225296 1, V-225292, V-225293, V-225294, V-225295, V-225296, V-225297, V-225298, V-225299, V-225300, V-225301, V-225302, V-225303, V-225305, V-225314, V-225314, V-225314, V-225314 15, V-225316, V-225317, V-225325, V-225326, V-225329, V-225337, V-225338, V-225339, V-225340, V-225341, V-225344, V-225345, V-225346, V-225347, V-225348, V-225349, V-225350, V-225351, V-225352, V-225353, V-225356, V-225367, V-225368, V-225369, V-225370, V-225371, V-225372, V-225373, V-225374, V-225375, V-225377, V-225378, V-225379, V-225380, V-225381, V-225382, V-225383, V-225384, V-225385, V-225386, V-225386, V-225386 389, V-225391, V-225393, V-225395, V-225397, V-225398, V-225400, V-225401, V-225402, V-225404, V-225405, V-225406, V-225407, V-225408, V-225409, V-225410, V-225410, V-225410 25411, V-225413, V-225414, V-225415, V-225441, V-225442, V-225443, V-225448, V-225452, V-225453, V-225454, V-225455, V-225456, V-225457, V-225458, V-225461, V-225463, V-225464, V-225469, V-225470, V-225471, V-225472, V-225474, V-225475, V-225477, V-225477, V-225477, V-225477, V-225477, V-225477 78, V-225486, V-225494, V-225500, V-225501, V-225502, V-225503, V-225504, V-225506, V-225508, V-225509, V-225510, V-225513, V-225515, V-225516, V-225517, V-225517, V-225517 518, V-225519, V-225520, V-225521, V-225522, V-225523, V-225524, V-225527, V-225528, V-225529, V-225530, V-225531, V-225532, V-225533, V-225534, V-225535, V-225535, V-225535, V-225535, V-225535, V-225535, V-225538, V-225539, V-225540, V-225541, V-225542, V-225543, V-225544, V-225545, V-225546, V-225548, V-225549, V-225550, V-225551, V-225553, V-225554, V-225555, V-225557, V-225558, V-225559, V-225560, V-225561, V-225562, V-225563, V-225564, V-225565, V-225566, V-225567, V-225568, V-225569, V-225570, V-225571, V-225572, V-225573, dan V-225574

- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 7

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-225223, V-225230, V-225235, dan V-225238

- Windows Firewall STIG Versi 2 Rilis 2

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

  V-241989, V-241990, V-241991, V-241993, V-241998, V-242003, V-242004, dan V-242005

- Internet Explorer 11 STIG Versi 2 Rilis 6

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

  V-223015, V-223017, V-223018, V-223019, V-223020, V-223021, V-223022, V-223023, V-223024, V-223025, V-223026, V-223027, V-223028, V-223029, V-223030, V-223031, V-223032, V-223033, V-223034, V-223035, V-223036, V-223037, V-223038, V-223039, V-223040, V-223041, V-223042, V-223043, V-223044, V-223045, V-223046, V-223048, V-223049, V-223050, V-223051, V-223052, V-223053, V-223053, V-223053 54, V-223055, V-223057, V-223058, V-223059, V-223060, V-223061, V-223062, V-223063, V-223064, V-223065, V-223066, V-223067, V-223068, V-223069, V-223070, V-223071, V-223072, V-223073, V-223073, V-223073, V-223073, V-223073 3074, V-223075, V-223076, V-223077, V-223079, V-223080, V-223081, V-223082, V-223083, V-223084, V-223085, V-223086, V-223087, V-223088, V-223089, V-223090, V-223091, V-223092, V-223093, V-223093, V-223094, V-223095, V-223096, V-223097, V-223098, V-223099, V-223100, V-223101, V-223102, V-223103, V-223104, V-223105, V-223106, V-223107, V-223108, V-223109, V-223110, V-223111, V-223112, V-223113, V-223114, V-223115, V-223116, V-223117, V-223118, V-223119, V-223120, V-223121, V-223123, V-223126, V-223127, V-223128, V-223129, V-223130, V-223130, V-223130 131, V-223132, V-223133, V-223134, V-223135, V-223136, V-223137, V-223138, V-223139, V-223140, V-223141, V-223142, V-223143, V-223144, V-223145, V-223146, V-223147, V-223148, V-223149, V-250540, dan V-250540 250541

- Microsoft Edge STIG Versi 2 Rilis 4 (hanya Windows Server 2022 dan 2025)

  V-235720, V-235721, V-235723, V-235724, V-235725, V-235726, V-235728, V-235729, V-235730, V-235732, V-235733, V-235734, V-235735, V-235736, V-235738, V-235739, V-235740, V-235741, V-235742, V-235743, V-235744, V-235745, V-235746, V-235747, V-235748, V-235749, V-235750, V-235754, V-235756, V-235761, V-235763, V-235764, V-235766, V-235767, V-235768, V-235768, V-235768, V-235768 69, V-235770, V-235771, V-235772, V-235773, V-235774, dan V-246736

- Microsoft Defender STIG Versi 2 Rilis 7

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori III (Rendah), ditambah:

V-213427, V-213429, V-213430, V-213431, V-213432, V-213433, V-213434, V-213435, V-213436, V-213437, V-213438, V-213439, V-213440, V-213441, V-213442, V-213442, V-213442 43, V-213444, V-213445, V-213446, V-213447, V-213448, V-213449, V-213450, V-213451, V-213454, V-213455, V-213456, V-213457, V-213458, V-213459, V-213460, V-213460, V-213460 461, V-213462, V-213463, V-213464, V-213465, V-213466, V-278647, V-278648, V-278649, V-278650, V-278651, V-278652, V-278654, V-278655, V-278656, V-278658, V-278658, V-278659, V-278659, V-278659, V-278659, V-278660, V-278661, V-278662, V-278668, V-278669, V-278672, V-278674, V-278675, V-278676, V-278677, V-278678, V-278679, V-278680, dan V-278863

Windows STIG Tinggi (Kategori I)

Daftar berikut berisi pengaturan STIG yang didukung Amazon EC2 ke infrastruktur Anda. Jika setelan yang didukung tidak berlaku untuk infrastruktur Anda, Amazon EC2 melewatkan setelan itu, dan melanjutkan. Misalnya, beberapa pengaturan pengerasan STIG mungkin tidak berlaku untuk server mandiri. Kebijakan khusus organisasi juga dapat memengaruhi pengaturan mana yang berlaku, seperti persyaratan bagi administrator untuk meninjau setelan dokumen.

Untuk daftar lengkap Windows STIGs, lihat Perpustakaan STIGs Dokumen. Untuk informasi tentang cara melihat daftar lengkap, lihat Alat Melihat STIG.

> **Note**
>
> Kategori Windows STIG High mencakup semua pengaturan pengerasan STIG terdaftar yang berlaku untuk kategori Windows STIG Medium dan Low, selain pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori I.

- Windows Server 2025 STIG Versi 1 Rilis 1

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

  V-278040, V-278099, V-278100, V-278101, V-278121, V-278125, V-278128, V-278196, V-278215, V-278216, V-278217, V-278219, V-278225, V-278242, V-278246, dan V-278250
- Windows Server 2022 STIG Versi 2 Rilis 7

Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-254250, V-254293, V-254352, V-254353, V-254354, V-254374, V-254378, V-254381, V-254446, V-254466, V-254467, V-254469, V-254474, V-254475, V-254492, V-254496, dan V-254500

- Windows Server 2019 STIG Versi 3 Rilis 7

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

  V-205653, V-205654, V-205663, V-205711, V-205713, V-205724, V-205725, V-205750, V-205753, V-205757, V-205802, V-205804, V-205805, V-205806, V-205849, V-205908, V-205914, dan V-205914 19

- Windows Server 2016 STIG Versi 2 Rilis 10

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

  V-224831, V-224874, V-224932, V-224933, V-224934, V-224954, V-224958, V-224961, V-225025, V-225045, V-225046, V-225048, V-225053, V-225054, V-225071, V-225079, dan V-225079 25091

- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

  V-225274, V-225354, V-225364, V-225365, V-225366, V-225390, V-225396, V-225399, V-225444, V-225449, V-225491, V-225492, V-225493, V-225496, V-225497, V-225498, V-225505, V-225507, V-225547, V-225552, dan V-225556

- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 7

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah) untuk Microsoft .NET Framework. Tidak ada pengaturan STIG tambahan yang berlaku untuk kerentanan Kategori I.

- Windows Firewall STIG Versi 2 Rilis 2

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

V-241992, V-241997, dan V-242002

- Internet Explorer 11 STIG Versi 2 Rilis 6

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

  V-252910

- Microsoft Edge STIG Versi 2 Rilis 4 (hanya Windows Server 2022 dan 2025)

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

  V-235758 dan V-235759

- Microsoft Defender STIG Versi 2 Rilis 7

  Termasuk semua pengaturan pengerasan STIG yang didukung Amazon EC2 untuk kerentanan Kategori II dan III (Sedang dan Rendah), ditambah:

  V-213426, V-213428, V-213452, dan V-213453

## Unduhan skrip pengerasan STIG

Amazon mengelompokkan skrip pengerasan STIG bersama-sama ke dalam bundel terkait sistem operasi untuk setiap rilis. Bundel adalah file arsip yang sesuai untuk sistem operasi target tempat mereka mengunduh dan menjalankannya. Bundel skrip Linux disimpan sebagai file TAR (ekstensi file.tgz). Bundel skrip Windows disimpan sebagai file ZIP (ekstensi file.zip).

Amazon menyimpan bundel skrip di `STIG` bucket EC2 Windows S3 di masing-masing. Wilayah AWS Bundel Linux tidak memiliki bucket terpisah. Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.

Topik

- Detail bundel unduhan STIG
- Riwayat versi Linux STIG
- Riwayat versi Windows STIG

## Detail bundel unduhan STIG

> **Important**
>
> Dengan beberapa pengecualian, skrip pengerasan STIG yang diunduh dokumen Systems Manager tidak menginstal paket pihak ketiga. Jika paket pihak ketiga sudah diinstal pada instance atau `InstallPackage` parameter disetel keyes, STIG terkait yang didukung Amazon EC2 untuk paket tersebut akan diterapkan.

Saat Anda menjalankan perintah berikut dari AWS CLI, Amazon S3 mengunduh file bundel skrip pengerasan STIG terbaru dari bucket.

```
aws s3 cp s3://aws-windows-downloads-region/STIG/operating system/Latest/bundle-
name destination-directory
```

Contoh: Unduh ke direktori sementara

Contoh ini menunjukkan bundel Linux yang diunduh ke `/tmp` direktori

```
aws s3 cp s3://aws-windows-downloads-us-east-1/STIG/Linux/
Latest/LinuxAWSConfigureSTIG.tgz /tmp
```

Pola dan contoh untuk jalur penyimpanan file unduhan dan nama file bundel adalah sebagai berikut:

Unduh jalur penyimpanan file

`s3://aws-windows-downloads-<region>/STIG/<operating system>/Latest/<bundle file name>`

Unduh variabel jalur

region

Wilayah AWS (Setiap Wilayah memiliki keranjang unduhannya sendiri.)

operating system

Platform sistem operasi dari instance di mana STIGs diterapkan - baik `Linux` atau`Windows`.

bundle file name

Formatnya adalah*<os bundle name>*. *<file extension>*.

os bundle name

Awalan nama standar untuk bundel sistem operasi adalah salah satu `LinuxAWSConfigureSTIG` atau`AWSConfigureSTIG`. Untuk mempertahankan kompatibilitas mundur, unduhan untuk Windows tidak menyertakan awalan platform.

file extension

Format file terkompresi `tgz` (Linux) atau `zip` (Windows).

Contoh nama file bundel

- `LinuxAWSConfigureSTIG.tgz`
- `AWSConfigureSTIG.zip`

## Riwayat versi Linux STIG

Bagian ini mencatat riwayat versi untuk bundel skrip Linux yang diperbarui setiap tiga bulan. Untuk melihat perubahan dan versi yang diterbitkan selama seperempat, pilih judul untuk memperluas informasi. Jika tidak ada perubahan untuk kuartal tersebut, Anda akan melihatnya tercermin dalam judul.

Perubahan Q1 2026 - 04/07/2026:

Menambahkan dukungan untuk sistem operasi RHEL 10 dan memperbarui versi STIG berikut dan menerapkan STIGS untuk rilis kuartal pertama 2026 untuk semua tingkat kepatuhan (): low/medium/high

STIG-Build-Linux versi 1.0.x

- RHEL 8 STIG Versi 2 Rilis 6
- RHEL 9 STIG Versi 2 Rilis 7
- RHEL 10 STIG Versi 1 Rilis 1
- Amazon Linux 2023 STIG Versi 1 Rilis 2
- SLES 12 STIG Versi 3 Rilis 4
- SLES 15 STIG Versi 2 Rilis 6
- Ubuntu 20.04 STIG Versi 2 Rilis 4
- Ubuntu 22.04 STIG Versi 2 Rilis 7

- Ubuntu 24.04 STIG Versi 1 Rilis 4

Perubahan Q3 2025 - 09/04/2025:

Menambahkan dukungan untuk sistem operasi SUSE Linux Enterprise Server (SLES) dan Amazon Linux 2023. Memperbarui versi STIG berikut dan menerapkan STIGS untuk rilis kuartal ketiga 2025 untuk semua tingkat kepatuhan (): low/medium/high

STIG-Build-Linux versi 1.0.x

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 4
- RHEL 9 STIG Versi 2 Rilis 5
- Amazon Linux 2023 STIG Versi 1 Rilis 1
- SLES 12 STIG Versi 3 Rilis 3
- SLES 15 STIG Versi 2 Rilis 5
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 3
- Ubuntu 22.04 STIG Versi 2 Rilis 5
- Ubuntu 24.04 STIG Versi 1 Rilis 2

Perubahan Q2 2025 - 26/06/2025:

Memperbarui versi STIG berikut, menerapkan STIGS untuk rilis kuartal kedua 2025:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 3
- RHEL 9 STIG Versi 2 Rilis 4
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 2
- Ubuntu 22.04 STIG Versi 2 Rilis 4
- Ubuntu 24.04 STIG Versi 1 Rilis 1

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 3
- RHEL 9 STIG Versi 2 Rilis 4
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 2
- Ubuntu 22.04 STIG Versi 2 Rilis 4
- Ubuntu 24.04 STIG Versi 1 Rilis 1

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 3
- RHEL 9 STIG Versi 2 Rilis 4
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 2
- Ubuntu 22.04 STIG Versi 2 Rilis 4
- Ubuntu 24.04 STIG Versi 1 Rilis 1

Perubahan Q1 2025 - 04/11/2025:

Memperbarui versi STIG berikut, menerapkan STIGS untuk rilis kuartal pertama 2025, dan menambahkan dukungan untuk Ubuntu 24.04:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 2
- RHEL 9 STIG Versi 2 Rilis 3
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 2
- Ubuntu 22.04 STIG Versi 2 Rilis 3
- Ubuntu 24.04 STIG Versi 1 Rilis 1

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 2
- RHEL 9 STIG Versi 2 Rilis 3
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 2
- Ubuntu 22.04 STIG Versi 2 Rilis 3
- Ubuntu 24.04 STIG Versi 1 Rilis 1

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 2
- RHEL 9 STIG Versi 2 Rilis 3
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 2
- Ubuntu 22.04 STIG Versi 2 Rilis 3
- Ubuntu 24.04 STIG Versi 1 Rilis 1

Perubahan Q4 2024 - 12/10/2024:

Memperbarui versi STIG berikut, menerapkan STIGS untuk rilis kuartal keempat 2024, dan menambahkan informasi tentang dua parameter input baru untuk komponen Linux:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 1
- RHEL 9 STIG Versi 2 Rilis 2
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 1
- Ubuntu 22.04 STIG Versi 2 Rilis 2

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 1
- RHEL 9 STIG Versi 2 Rilis 2
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 1
- Ubuntu 22.04 STIG Versi 2 Rilis 2

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 15
- RHEL 8 STIG Versi 2 Rilis 1
- RHEL 9 STIG Versi 2 Rilis 2
- Ubuntu 18.04 STIG Versi 2 Rilis 15
- Ubuntu 20.04 STIG Versi 2 Rilis 1
- Ubuntu 22.04 STIG Versi 2 Rilis 2

Perubahan Q3 2024 - 10/04/2024 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Linux STIGS untuk rilis kuartal ketiga 2024.

Perubahan Q2 2024 - 05/10/2024:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal kedua 2024. Juga menambahkan dukungan untuk RHEL 9, CentOS Stream 9, dan Ubuntu 22.04, sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 14
- RHEL 8 STIG Versi 1 Rilis 14
- RHEL 9 STIG Versi 1 Rilis 3
- Ubuntu 18.04 STIG Versi 2 Rilis 14
- Ubuntu 20.04 STIG Versi 1 Rilis 12
- Ubuntu 22.04 STIG Versi 1 Rilis 1

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 14
- RHEL 8 STIG Versi 1 Rilis 14
- RHEL 9 STIG Versi 1 Rilis 3
- Ubuntu 18.04 STIG Versi 2 Rilis 14
- Ubuntu 20.04 STIG Versi 1 Rilis 12
- Ubuntu 22.04 STIG Versi 1 Rilis 1

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 14
- RHEL 8 STIG Versi 1 Rilis 14
- RHEL 9 STIG Versi 1 Rilis 3
- Ubuntu 18.04 STIG Versi 2 Rilis 14
- Ubuntu 20.04 STIG Versi 1 Rilis 12
- Ubuntu 22.04 STIG Versi 1 Rilis 1

Perubahan Q1 2024 - 02/06/2024:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal pertama 2024 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 14
- RHEL 8 STIG Versi 1 Rilis 13
- Ubuntu 18.04 STIG Versi 2 Rilis 13
- Ubuntu 20.04 STIG Versi 1 Rilis 11

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 14
- RHEL 8 STIG Versi 1 Rilis 13
- Ubuntu 18.04 STIG Versi 2 Rilis 13
- Ubuntu 20.04 STIG Versi 1 Rilis 11

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 14
- RHEL 8 STIG Versi 1 Rilis 13
- Ubuntu 18.04 STIG Versi 2 Rilis 13
- Ubuntu 20.04 STIG Versi 1 Rilis 11

Perubahan Q4 2023 - 12/07/2023:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal keempat 2023 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 13
- RHEL 8 STIG Versi 1 Rilis 12
- Ubuntu 18.04 STIG Versi 2 Rilis 12
- Ubuntu 20.04 STIG Versi 1 Rilis 10

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 13
- RHEL 8 STIG Versi 1 Rilis 12
- Ubuntu 18.04 STIG Versi 2 Rilis 12
- Ubuntu 20.04 STIG Versi 1 Rilis 10

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 13
- RHEL 8 STIG Versi 1 Rilis 12
- Ubuntu 18.04 STIG Versi 2 Rilis 12
- Ubuntu 20.04 STIG Versi 1 Rilis 10

Perubahan Q3 2023 - 10/04/2023:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal ketiga 2023 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 12
- RHEL 8 STIG Versi 1 Rilis 11
- Ubuntu 18.04 STIG Versi 2 Rilis 11
- Ubuntu 20.04 STIG Versi 1 Rilis 9

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 12
- RHEL 8 STIG Versi 1 Rilis 11
- Ubuntu 18.04 STIG Versi 2 Rilis 11
- Ubuntu 20.04 STIG Versi 1 Rilis 9

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 12
- RHEL 8 STIG Versi 1 Rilis 11
- Ubuntu 18.04 STIG Versi 2 Rilis 11
- Ubuntu 20.04 STIG Versi 1 Rilis 9

Perubahan Q2 2023 - 05/03/2023:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal kedua 2023 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 11
- RHEL 8 STIG Versi 1 Rilis 10
- Ubuntu 18.04 STIG Versi 2 Rilis 11
- Ubuntu 20.04 STIG Versi 1 Rilis 8

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 11

- RHEL 8 STIG Versi 1 Rilis 10
- Ubuntu 18.04 STIG Versi 2 Rilis 11
- Ubuntu 20.04 STIG Versi 1 Rilis 8

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 11
- RHEL 8 STIG Versi 1 Rilis 10
- Ubuntu 18.04 STIG Versi 2 Rilis 11
- Ubuntu 20.04 STIG Versi 1 Rilis 8

Perubahan Q1 2023 - 27/03/2023:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal pertama 2023 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 10
- RHEL 8 STIG Versi 1 Rilis 9
- Ubuntu 18.04 STIG Versi 2 Rilis 10
- Ubuntu 20.04 STIG Versi 1 Rilis 7

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 10
- RHEL 8 STIG Versi 1 Rilis 9
- Ubuntu 18.04 STIG Versi 2 Rilis 10
- Ubuntu 20.04 STIG Versi 1 Rilis 7

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 10
- RHEL 8 STIG Versi 1 Rilis 9
- Ubuntu 18.04 STIG Versi 2 Rilis 10

- Ubuntu 20.04 STIG Versi 1 Rilis 7

2022 Perubahan Q4 - 02/01/2023:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal keempat 2022 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 9
- RHEL 8 STIG Versi 1 Rilis 8
- Ubuntu 18.04 STIG Versi 2 Rilis 9
- Ubuntu 20.04 STIG Versi 1 Rilis 6

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 9
- RHEL 8 STIG Versi 1 Rilis 8
- Ubuntu 18.04 STIG Versi 2 Rilis 9
- Ubuntu 20.04 STIG Versi 1 Rilis 6

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 9
- RHEL 8 STIG Versi 1 Rilis 8
- Ubuntu 18.04 STIG Versi 2 Rilis 9
- Ubuntu 20.04 STIG Versi 1 Rilis 6

2022 Perubahan Q3 - 30/09/2022 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Linux STIGS untuk rilis kuartal ketiga 2022.

2022 Perubahan Q2 - 08/02/2022:

Memperkenalkan dukungan Ubuntu, versi STIG yang diperbarui, dan menerapkan STIGS untuk rilis kuartal kedua 2022 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 7
- RHEL 8 STIG Versi 1 Rilis 6
- Ubuntu 18.04 STIG Versi 2 Rilis 6 (baru)
- Ubuntu 20.04 STIG Versi 1 Rilis 4 (baru)

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 7
- RHEL 8 STIG Versi 1 Rilis 6
- Ubuntu 18.04 STIG Versi 2 Rilis 6 (baru)
- Ubuntu 20.04 STIG Versi 1 Rilis 4 (baru)

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 7
- RHEL 8 STIG Versi 1 Rilis 6
- Ubuntu 18.04 STIG Versi 2 Rilis 6 (baru)
- Ubuntu 20.04 STIG Versi 1 Rilis 4 (baru)

2022 Perubahan Q1 - 26/04/2022:

Difaktorkan ulang untuk menyertakan dukungan yang lebih baik untuk wadah. Menggabungkan AL2 skrip sebelumnya dengan RHEL 7. Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal pertama 2022 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 6
- RHEL 8 STIG Versi 1 Rilis 5

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 6
- RHEL 8 STIG Versi 1 Rilis 5

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 6
- RHEL 8 STIG Versi 1 Rilis 5

2021 Perubahan Q4 - 12/20/2021:

Versi STIG yang diperbarui, dan menerapkan STIGS untuk rilis kuartal keempat 2021 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 5
- RHEL 8 STIG Versi 1 Rilis 4

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 5
- RHEL 8 STIG Versi 1 Rilis 4

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 5
- RHEL 8 STIG Versi 1 Rilis 4

2021 Perubahan Q3 - 09/30/2021:

Versi STIG yang diperbarui, dan menerapkan STIGS untuk rilis kuartal ketiga 2021 sebagai berikut:

Linux STIG Rendah (Kategori III)

- RHEL 7 STIG Versi 3 Rilis 4
- RHEL 8 STIG Versi 1 Rilis 3

Linux STIG Medium (Kategori II)

- RHEL 7 STIG Versi 3 Rilis 4

- RHEL 8 STIG Versi 1 Rilis 3

Linux STIG Tinggi (Kategori I)

- RHEL 7 STIG Versi 3 Rilis 4
- RHEL 8 STIG Versi 1 Rilis 3

## Riwayat versi Windows STIG

Bagian ini mencatat riwayat versi untuk bundel skrip Windows yang diperbarui setiap tiga bulan. Untuk melihat perubahan dan versi yang diterbitkan selama seperempat, pilih judul untuk memperluas informasi. Jika tidak ada perubahan untuk kuartal tersebut, Anda akan melihatnya tercermin dalam judul.

Perubahan Q1 2026 - 04/07/2026:

Menambahkan dukungan untuk sistem operasi Windows Server 2025 dan memperbarui versi STIG berikut dan menerapkan STIGS untuk rilis kuartal pertama 2026 untuk semua tingkat kepatuhan (): low/medium/high

STIG-Build-Windows versi 1.0.x

- Windows Server 2025 STIG Versi 1 Rilis 1
- Windows Server 2022 STIG Versi 2 Rilis 7
- Windows Server 2019 STIG Versi 3 Rilis 7
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 7
- Windows Firewall STIG Versi 2 Rilis 2
- Internet Explorer 11 STIG Versi 2 Rilis 6
- Microsoft Edge STIG Versi 2 Rilis 4 (hanya Windows Server 2022 dan 2025)

Perubahan Q4 2025 - 12/10/2025 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal keempat 2025.

Perubahan Q3 2025 - 9/04/2025 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal ketiga 2025.

Perubahan Q2 2025 - 26/06/2025:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis Q2 2025 sebagai berikut:

Windows STIG Rendah (Kategori III)

- Windows Server 2022 STIG Versi 2 Rilis 4
- Windows Server 2019 STIG Versi 3 Rilis 4
- Windows Server 2016 STIG Versi 2 Rilis 10
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 6
- Windows Firewall STIG Versi 2 Rilis 2
- Internet Explorer 11 STIG Versi 2 Rilis 5
- Microsoft Edge STIG Versi 2 Rilis 2 (hanya Windows Server 2022)

Windows STIG Medium (Kategori II)

- Windows Server 2022 STIG Versi 2 Rilis 4
- Windows Server 2019 STIG Versi 3 Rilis 4
- Windows Server 2016 STIG Versi 2 Rilis 10
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 6
- Windows Firewall STIG Versi 2 Rilis 2
- Internet Explorer 11 STIG Versi 2 Rilis 5
- Microsoft Edge STIG Versi 2 Rilis 2 (hanya Windows Server 2022)
- Defender STIG Versi 2 Rilis 4

Windows STIG Tinggi (Kategori I)

- Windows Server 2022 STIG Versi 2 Rilis 4
- Windows Server 2019 STIG Versi 3 Rilis 4
- Windows Server 2016 STIG Versi 2 Rilis 10
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5

- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 6
- Windows Firewall STIG Versi 2 Rilis 2
- Internet Explorer 11 STIG Versi 2 Rilis 5
- Microsoft Edge STIG Versi 2 Rilis 2 (hanya Windows Server 2022)
- Defender STIG Versi 2 Rilis 4

Perubahan Q1 2025 - 05/04/2025:

STIGS yang diperbarui untuk Internet Explorer 11 STIG Versi 2 Rilis 5 untuk semua komponen STIG untuk rilis kuartal pertama 2025.

Perubahan Q4 2024 - 12/10/2024:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis Q4 2024 sebagai berikut:

Windows STIG Rendah (Kategori III)

- Windows Server 2022 STIG Versi 2 Rilis 2
- Windows Server 2019 STIG Versi 3 Rilis 2
- Windows Server 2016 STIG Versi 2 Rilis 9
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 2
- Windows Firewall STIG Versi 2 Rilis 2
- Internet Explorer 11 STIG Versi 2 Rilis 5
- Microsoft Edge STIG Versi 2 Rilis 2 (hanya Windows Server 2022)

Windows STIG Medium (Kategori II)

- Windows Server 2022 STIG Versi 2 Rilis 2
- Windows Server 2019 STIG Versi 3 Rilis 2
- Windows Server 2016 STIG Versi 2 Rilis 9
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 2

- Windows Firewall STIG Versi 2 Rilis 2
- Internet Explorer 11 STIG Versi 2 Rilis 5
- Microsoft Edge STIG Versi 2 Rilis 2 (hanya Windows Server 2022)
- Defender STIG Versi 2 Rilis 4

Windows STIG Tinggi (Kategori I)

- Windows Server 2022 STIG Versi 2 Rilis 2
- Windows Server 2019 STIG Versi 3 Rilis 2
- Windows Server 2016 STIG Versi 2 Rilis 9
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 2
- Windows Firewall STIG Versi 2 Rilis 2
- Internet Explorer 11 STIG Versi 2 Rilis 5
- Microsoft Edge STIG Versi 2 Rilis 2 (hanya Windows Server 2022)
- Defender STIG Versi 2 Rilis 4

Perubahan Q3 2024 - 10/04/2024 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal ketiga 2024.

Perubahan Q2 2024 - 05/10/2024 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal kedua 2024.

Perubahan Q1 2024 - 23/02/2024 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal pertama 2024.

Perubahan Q4 2023 - 12/07/2023 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal keempat 2023.

Perubahan Q3 2023 - 10/04/2023 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal ketiga 2023.

Perubahan Q2 2023 - 05/03/2023 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal kedua 2023.

Perubahan Q1 2023 - 27/03/2023 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal pertama 2023.

2022 Perubahan Q4 - 02/01/2023:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis Q4 2022 sebagai berikut:

Windows STIG Rendah (Kategori III)

- Windows Server 2022 STIG Versi 1 Rilis 1
- Windows Server 2019 STIG Versi 2 Rilis 5
- Windows Server 2016 STIG Versi 2 Rilis 5
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 2
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 2 Rilis 3
- Microsoft Edge STIG Versi 1 Rilis 6 (hanya Windows Server 2022)

Windows STIG Medium (Kategori II)

- Windows Server 2022 STIG Versi 1 Rilis 1
- Windows Server 2019 STIG Versi 2 Rilis 5
- Windows Server 2016 STIG Versi 2 Rilis 5
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 2
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 2 Rilis 3
- Microsoft Edge STIG Versi 1 Rilis 6 (hanya Windows Server 2022)
- Defender STIG Versi 2 Rilis 4 (hanya Windows Server 2022)

Windows STIG Tinggi (Kategori I)

- Windows Server 2022 STIG Versi 1 Rilis 1
- Windows Server 2019 STIG Versi 2 Rilis 5
- Windows Server 2016 STIG Versi 2 Rilis 5
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 5
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 2
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 2 Rilis 3
- Microsoft Edge STIG Versi 1 Rilis 6 (hanya Windows Server 2022)
- Defender STIG Versi 2 Rilis 4 (hanya Windows Server 2022)

2022 Perubahan Q3 - 30/09/2022 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal ketiga 2022.

2022 Perubahan Q2 - 08/02/2022:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis Q2 2022.

Windows STIG Rendah (Kategori III)

- Windows Server 2019 STIG Versi 2 Rilis 4
- Windows Server 2016 STIG Versi 2 Rilis 4
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 3
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 1 Rilis 19

Windows STIG Medium (Kategori II)

- Windows Server 2019 STIG Versi 2 Rilis 4
- Windows Server 2016 STIG Versi 2 Rilis 4
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 3

- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 1 Rilis 19

Windows STIG Tinggi (Kategori I)

- Windows Server 2019 STIG Versi 2 Rilis 4
- Windows Server 2016 STIG Versi 2 Rilis 4
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 3
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 1 Rilis 19

2022 Perubahan Q1 - 08/02/2022 (tidak ada perubahan):

Tidak ada perubahan untuk komponen Windows STIGS untuk rilis kuartal pertama 2022.

2021 Perubahan Q4 - 12/20/2021:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal keempat 2021.

Windows STIG Rendah (Kategori III)

- Windows Server 2019 STIG Versi 2 Rilis 3
- Windows Server 2016 STIG Versi 2 Rilis 3
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 3
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 1 Rilis 19

Windows STIG Medium (Kategori II)

- Windows Server 2019 STIG Versi 2 Rilis 3
- Windows Server 2016 STIG Versi 2 Rilis 3
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 3

- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 1 Rilis 19

Windows STIG Tinggi (Kategori I)

- Windows Server 2019 STIG Versi 2 Rilis 3
- Windows Server 2016 STIG Versi 2 Rilis 3
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 3
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 2 Rilis 1
- Internet Explorer 11 STIG Versi 1 Rilis 19

2021 Perubahan Q3 - 09/30/2021:

Versi STIG yang diperbarui dan menerapkan STIGS untuk rilis kuartal ketiga 2021.

Windows STIG Rendah (Kategori III)

- Windows Server 2019 STIG Versi 2 Rilis 2
- Windows Server 2016 STIG Versi 2 Rilis 2
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 2
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 1 Rilis 7
- Internet Explorer 11 STIG Versi 1 Rilis 19

Windows STIG Medium (Kategori II)

- Windows Server 2019 STIG Versi 2 Rilis 2
- Windows Server 2016 STIG Versi 2 Rilis 2
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 2
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 1 Rilis 7

- Internet Explorer 11 STIG Versi 1 Rilis 19

Windows STIG Tinggi (Kategori I)

- Windows Server 2019 STIG Versi 2 Rilis 2
- Windows Server 2016 STIG Versi 2 Rilis 2
- Windows Server 2012 R2 MS STIG Versi 3 Rilis 2
- Microsoft .NET Framework 4.0 STIG Versi 2 Rilis 1
- Windows Firewall STIG Versi 1 Rilis 7
- Internet Explorer 11 STIG Versi 1 Rilis 19

## Gunakan AWS Systems Manager untuk menerapkan pengaturan STIG ke instans Anda

Anda dapat menggunakan dokumen perintah `AWSEC2-ConfigureSTIG` Systems Manager untuk menerapkan pengaturan STIG ke instans EC2 yang ada. Anda harus menjalankan dokumen perintah dari contoh yang diperbarui. Dokumen perintah menerapkan pengaturan yang sesuai berdasarkan sistem operasi dan konfigurasi instance di mana ia berjalan.

Halaman ini berisi rincian tentang dokumen `AWSEC2-ConfigureSTIG` perintah, termasuk parameter input dan cara menjalankannya di konsol Systems Manager atau dengan `send-command` di file AWS CLI.

### AWSEC2-ConfigureSTIGparameter masukan

Anda dapat memberikan parameter input berikut untuk menentukan bagaimana dokumen perintah harus menerapkan pengaturan STIG ke instance Anda.

Level (string, diperlukan)

Tentukan kategori keparahan STIG untuk diterapkan. Nilai-nilai yang valid meliputi:

- Tinggi
- Sedang
- Rendah

Jika Anda tidak menentukan nilai default sistem. `High`

InstallPackages(string, opsional — hanya Linux)

Jika nilainyaNo, skrip tidak menginstal paket perangkat lunak tambahan apa pun. Jika nilainyaYes, skrip menginstal paket perangkat lunak tambahan yang diperlukan untuk kepatuhan maksimum. Nilai default-nya adalah No.

SetDoDConsentBanner (string, opsional — hanya Linux)

Jika nilainyaNo, spanduk persetujuan DoD tidak ditampilkan saat Anda melampirkan ke instance yang memiliki skrip STIG Linux diinstal. Jika nilainyaYes, spanduk persetujuan DoD ditampilkan sebelum Anda masuk saat Anda melampirkan ke instance yang memiliki salah satu skrip STIG Linux diinstal. Anda harus mengakui spanduk sebelum Anda dapat masuk. Nilai default-nya adalah No.

Untuk contoh spanduk persetujuan, lihat Pemberitahuan Privasi dan Persetujuan Departemen Pertahanan Penafian yang muncul saat Anda mengakses situs web Layanan Dokumen DLA.

## Jalankan dokumen AWSEC2-ConfigureSTIG perintah

Untuk menjalankan dokumen `AWSEC2-ConfigureSTIG`, ikuti langkah-langkah untuk lingkungan pilihan Anda.

Console

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Pilih Jalankan Perintah dari panel navigasi. Ini menunjukkan daftar perintah yang sedang berjalan di akun Anda, jika berlaku.
3. Pilih Jalankan perintah. Ini membuka dialog Run a command dan menampilkan daftar dokumen perintah yang dapat Anda akses.
4. Pilih `AWSEC2-ConfigureSTIG` dari daftar dokumen perintah. Untuk merampingkan hasil, Anda dapat memasukkan semua atau sebagian dari nama dokumen. Anda juga dapat memfilter berdasarkan pemilik, berdasarkan jenis platform, atau dengan tanda.

   Saat Anda memilih dokumen perintah, detail terisi di bawah daftar.
5. Pilih `Default version at runtime` dari daftar Versi dokumen.
6. Konfigurasikan parameter Command untuk menentukan bagaimana `AWSEC2-ConfigureSTIG` akan menginstal paket skrip dan menjalankannya untuk memperbarui instance Anda. Untuk detail parameter, lihat AWSEC2-ConfigureSTIGparameter masukan.

7. Untuk pemilihan target, tentukan tanda atau pilih instans secara manual untuk mengidentifikasi instans untuk menjalankan operasi ini.

   > **Note**
   >
   > Jika Anda memilih instans secara manual, dan instans yang ingin Anda lihat tidak disertakan dalam daftar, lihat Di Mana Instans Saya? untuk kiat pemecahan masalah.

8. Untuk parameter tambahan guna menentukan perilaku Run Command Systems Manager seperti Kontrol laju, masukkan nilai seperti yang dijelaskan dalam Menjalankan perintah dari konsol.
9. Pilih Jalankan.

   Jika berhasil, dokumen perintah menginstal skrip dan mengkonfigurasi instance Anda. Jika pelaksanaan perintah gagal, lihat output perintah Systems Manager untuk detail tentang alasan pelaksanaan tersebut gagal.

AWS CLI

Contoh 1: Jalankan dengan nilai default

Jalankan perintah berikut untuk menginstal skrip STIG dan menjalankannya dengan nilai default. Untuk informasi selengkapnya tentang parameter input, lihatAWSEC2-ConfigureSTIGparameter masukan.

```
aws ssm send-command \
 --document-name "AWSEC2-ConfigureSTIG" \
 --instance-ids "i-1234567890abcdef0"'
```

Contoh 2: Konfigurasikan pengaturan STIG tingkat Menengah pada instans Anda

Jalankan perintah berikut untuk menginstal skrip STIG dan menjalankannya dengan parameter `Level` input diatur ke`Medium`. Untuk informasi selengkapnya tentang parameter input, lihatAWSEC2-ConfigureSTIGparameter masukan.

```
aws ssm send-command \
 --document-name "AWSEC2-ConfigureSTIG" \
 --instance-ids "i-1234567890abcdef0"
 --parameters '{"Level":"Medium"}'
```

Jika berhasil, dokumen perintah menginstal skrip dan mengkonfigurasi instance Anda. Jika pelaksanaan perintah gagal, lihat output perintah untuk detail tentang alasan pelaksanaan tersebut gagal.

PowerShell

Contoh 1: Jalankan dengan nilai default

Jalankan perintah berikut untuk menginstal skrip STIG dan menjalankannya dengan nilai default. Untuk informasi selengkapnya tentang parameter input, lihatAWSEC2-ConfigureSTIGparameter masukan.

```
Send-SSMCommand -DocumentName "AWSEC2-ConfigureSTIG" -InstanceId
 "i-1234567890abcdef0"}
```

Contoh 2: Konfigurasikan pengaturan STIG tingkat Menengah pada instans Anda

Jalankan perintah berikut untuk menginstal skrip STIG dan menjalankannya dengan parameter `Level` input diatur ke`Medium`. Untuk informasi selengkapnya tentang parameter input, lihatAWSEC2-ConfigureSTIGparameter masukan.

```
Send-SSMCommand -DocumentName "AWSEC2-ConfigureSTIG" -InstanceId
 "i-1234567890abcdef0" -Parameter @{'Level'='Medium'}
```

Jika berhasil, dokumen perintah menginstal skrip dan mengkonfigurasi instance Anda. Jika pelaksanaan perintah gagal, lihat output perintah untuk detail tentang alasan pelaksanaan tersebut gagal.

## Sinkronisasi jam dan waktu presisi pada instans EC2 Anda

Referensi waktu yang konsisten dan akurat pada instans Amazon EC2 Anda sangat penting untuk banyak tugas dan proses server. Stempel waktu dalam log sistem memainkan peran penting dalam mengidentifikasi kapan masalah terjadi dan urutan kronologis peristiwa. Saat Anda menggunakan AWS CLI atau AWS SDK untuk membuat permintaan dari instans Anda, alat ini menandatangani permintaan atas nama Anda. Jika pengaturan tanggal dan waktu instans Anda tidak akurat, hal itu dapat menyebabkan perbedaan antara tanggal dalam tanda tangan dan tanggal permintaan, sehingga permintaan Anda dit AWS olak.

Untuk mengatasi aspek penting ini, Amazon menawarkan Layanan Amazon Time Sync, yang dapat diakses dari semua instans EC2 dan digunakan oleh berbagai Layanan AWS. Layanan

ini menggunakan armada jam referensi satelit dan atom di masing-masing Wilayah AWS untuk memberikan pembacaan waktu yang akurat dan terkini dari standar global Waktu Universal Terkoordinasi (UTC).

Untuk performa terbaik, sebaiknya gunakan Layanan Sinkronisasi Waktu Amazon lokal pada instans EC2 Anda. Untuk membuat cadangan ke Layanan Sinkronisasi Waktu Amazon lokal pada instans Anda, atau untuk menghubungkan sumber daya di luar Amazon EC2 ke Layanan Sinkronisasi Waktu Amazon, Anda dapat menggunakan Layanan Sinkronisasi Waktu Amazon publik yang terletak di `time.aws.com`. Layanan Amazon Time Sync publik, seperti Layanan Amazon Time Sync, secara otomatis menyebarkan setiap detik kabisat yang ditambahkan ke UTC. Layanan Sinkronisasi Waktu Amazon publik didukung secara global oleh armada jam referensi satelit dan atom kami di masing-masing jam. Wilayah AWS

## Stempel waktu paket perangkat keras

Anda dapat mengaktifkan stempel waktu paket perangkat keras pada instans Anda untuk menambahkan stempel waktu presisi nano detik 64-bit ke setiap paket jaringan yang masuk. Karena timestamp paket perangkat keras terjadi pada tingkat perangkat keras—sebelum paket mencapai kernel, soket, atau lapisan aplikasi—Anda melewati penundaan apa pun yang ditambahkan oleh stempel waktu perangkat lunak. Jam referensi yang mendasari untuk stempel waktu perangkat keras adalah jam perangkat keras Amazon Time Sync Service PTP.

### Manfaat

Timestamp paket perangkat keras memberikan manfaat berikut:

- Meningkatkan urutan peristiwa, yang juga dapat digunakan untuk menentukan urutan aktual di mana paket tiba di instans EC2 Anda, memastikan pemrosesan paket yang adil.
- Mengukur latensi jaringan satu arah.
- Meningkatkan kecepatan transaksi terdistribusi dengan presisi dan akurasi yang lebih tinggi dibandingkan dengan sebagian besar solusi lokal.

### Prasyarat dan konfigurasi

Untuk mengaktifkan stempel waktu paket perangkat keras, instans Anda harus memenuhi prasyarat berikut:

- Harus instance Linux.

- Memenuhi persyaratan untuk mendukung jam perangkat keras PTP.

Untuk petunjuk konfigurasi, lihat Hardware Packet Timestamping pada driver kernel Linux untuk halaman keluarga Elastic Network Adapter (ENA) di GitHub.

## Detik kabisat

Detik kabisat, diperkenalkan pada tahun 1972, adalah penyesuaian sesekali satu detik pada waktu UTC untuk memperhitungkan penyimpangan dalam rotasi bumi untuk mengakomodasi perbedaan antara Waktu Atom Internasional (TAI) dan waktu matahari (Ut1). Untuk mengelola detik kabisat atas nama pelanggan, kami merancang smearing detik kabisat dalam Layanan Amazon Time Sync. Untuk informasi selengkapnya, lihat Lihat Sebelum Anda Melompat — Detik Kabisat yang Akan Datang dan AWS.

Detik kabisat akan hilang, dan kami mendukung penuh keputusan yang dibuat pada Konferensi Umum ke-27 tentang Berat dan Ukuran untuk meninggalkan detik kabisat pada atau sebelum 2035.

Untuk mendukung transisi ini, kami masih berencana untuk smearing waktu peristiwa detik kabisat saat mengakses Layanan Amazon Time Sync melalui koneksi NTP lokal atau kolam NTP publik kami (`time.aws.com`). Namun, jam perangkat keras tidak menyediakan opsi waktu dengan smearing. Jika terjadi detik kabisat, jam perangkat keras PTP akan menambahkan detik kabisat mengikuti standar UTC. Leap-smeared dan sumber waktu kedua lompat sama dalam banyak kasus. Namun, karena keduanya berbeda selama peristiwa detik kabisat, kami tidak menyarankan penggunaan sumber waktu dengan smearing maupun tanpa smearing dalam konfigurasi klien waktu Anda selama peristiwa detik kabisat.

Topik

- Mengatur referensi waktu pada instans EC2 Anda untuk menggunakan Layanan Sinkronisasi Waktu Amazon lokal
- Mengatur referensi waktu pada instans EC2 Anda atau perangkat yang terhubung ke internet untuk menggunakan Layanan Sinkronisasi Waktu Amazon publik
- Bandingkan stempel waktu untuk instans Linux Anda
- Mengubah zona waktu instans Anda

Sumber daya terkait

- AWS Blog Komputasi: Sudah Tentang Waktu: Microsecond-Accurate Jam di Instans Amazon EC2
- AWS Blog Operasi & Migrasi Cloud: Kelola akurasi jam instans Amazon EC2 menggunakan Layanan Sinkronisasi Waktu Amazon dan Amazon CloudWatch - Bagian 1
- (Linux) https://chrony-project.org/

## Mengatur referensi waktu pada instans EC2 Anda untuk menggunakan Layanan Sinkronisasi Waktu Amazon lokal

Layanan Sinkronisasi Waktu Amazon menyediakan beberapa metode untuk instans Amazon EC2 Anda untuk menyinkronkan ke sumber waktu lokal. Pertama, instans Amazon EC2 apa pun dapat mencapai sumber waktu lokal melalui Network Time Protocol (NTP). Selain itu, Layanan Sinkronisasi Waktu Amazon yang disempurnakan menawarkan sumber waktu lokal dengan akurasi lebih tinggi untuk instans Amazon EC 2 yang didukung. Luncurkan instance yang didukung dalam grup penempatan dengan `precision-time` strategi untuk mencapai sumber NTP dengan akurasi yang lebih tinggi. Akhirnya, instance Linux yang diluncurkan dalam grup penempatan waktu presisi memiliki akses ke perangkat PTP Hardware Clock (PHC) dan kemampuan untuk mengambil cap waktu paket perangkat keras.

Instans Amazon EC2 apa pun memiliki akses ke sumber NTP lokal. Anda dapat mencapai sumber NTP melalui alamat IP link-local, yang membatasi lalu lintas ini ke dalam VPC Anda tanpa perlu perubahan konfigurasi VPC tertentu. AMI Anda mungkin sudah mengonfigurasi daemon sinkronisasi jam Anda untuk menggunakan sumber NTP lokal secara default. Sumber NTP ini tersedia melalui alamat IP berikut:

- IPv4: `169.254.169.123`
- IPv6: `fd00:ec2::123` (Hanya dapat diakses pada Nitro-based instance.)

Instans Amazon EC2 yang didukung memiliki akses ke Layanan Sinkronisasi Waktu Amazon yang disempurnakan. Untuk mengakses Layanan Sinkronisasi Waktu Amazon yang disempurnakan, luncurkan instance yang didukung dalam grup penempatan dengan `precision-time` strategi tersebut. Anda tidak perlu mengkonfigurasi instance Anda untuk mendapatkan manfaat dari peningkatan ini jika Anda menggunakan alamat IP link-lokal NTP. Sistem operasi apa pun dapat menggunakan peningkatan ini. Anda dapat memverifikasi bahwa Anda mendapat manfaat dari sumber NTP yang disempurnakan dengan menggunakan klien NTP pilihan Anda.

Linux-based AMI, berjalan pada keluarga instans yang didukung, memiliki opsi tambahan untuk sumber waktu dari perangkat PHC. Driver ENA membuat perangkat ini tersedia. Baik sumber NTP yang disempurnakan dan perangkat PHC menggunakan sumber waktu yang sangat akurat yang sama. Akses ke sumber waktu PHC dioptimalkan, yang mengarah ke sinkronisasi instans Amazon EC2 Anda yang lebih akurat.

Pertimbangan-pertimbangan

- Sumber waktu NTP menawarkan tampilan melompat dari skala waktu UTC, sedangkan PHC tidak mengotori waktu. Untuk informasi selengkapnya, lihat Detik kabisat.
- Hanya instans Linux yang memiliki akses ke jam perangkat keras PTP lokal. Instans Windows harus menggunakan NTP untuk mengakses Layanan Sinkronisasi Waktu Amazon yang disempurnakan.
- Ada batas 1024 paket per detik (PPS) untuk layanan yang menggunakan alamat link-local. Batas ini mencakup agregat permintaan Kueri DNS Resolver Route 53, permintaan Layanan Metadata Instance (IMDS), permintaan Amazon Time Sync Service Network Time Protocol (NTP), dan permintaan Layanan Lisensi Windows (untuk instans berbasis Microsoft Windows).

Daftar Isi


- Akses titik akhir IPv4 dari Layanan Sinkronisasi Waktu Amazon
- Akses titik akhir IPv6 dari Layanan Sinkronisasi Waktu Amazon
- Akses Layanan Sinkronisasi Waktu Amazon yang disempurnakan
- Akses Jam Perangkat Keras PTP (PHC)


## Akses titik akhir IPv4 dari Layanan Sinkronisasi Waktu Amazon

AMI Anda mungkin sudah mengonfigurasi Layanan Sinkronisasi Waktu Amazon secara default. Jika tidak, gunakan prosedur berikut untuk mengonfigurasi instans Anda agar menggunakan Layanan Sinkronisasi Waktu Amazon lokal melalui titik akhir IPv4.

Untuk bantuan memecahkan masalah, lihat Memecahkan masalah sinkronisasi NTP pada instans Linux atau Memec ahkan masalah waktu pada inst ans Windows.

### Amazon Linux

AL2023 dan versi terbaru Amazon Linux 2 dikonfigurasi untuk menggunakan titik akhir IPv4 Layanan Sinkronisasi Waktu Amazon secara default. Jika Anda mengonfirmasi bahwa instance Anda sudah dikonfigurasi, Anda dapat melewati prosedur berikut.

Untuk memverifikasi bahwa chrony dikonfigurasi untuk menggunakan titik akhir IPv4

Jalankan perintah berikut. Dalam output, baris yang dimulai dengan `^*` menunjukkan sumber waktu yang disukai.

```
[ec2-user ~]$ chronyc sources -v | grep -F ^*
```

```
^* 169.254.169.123               3   4   377    13  -4325ns[-9201ns] +/-  401us
```

Untuk mengkonfigurasi chrony agar terhubung ke titik akhir IPv4 pada versi Amazon Linux 2 yang lebih lama

1. Hubungkan ke instans Anda dan hapus instalasi layanan NTP.

   ```
   [ec2-user ~]$ sudo yum erase 'ntp*'
   ```

2. Instal paket `chrony`.

   ```
   [ec2-user ~]$ sudo yum install chrony
   ```

3. Buka `/etc/chrony.conf` file menggunakan editor teks (sepertivim ataunano ). Tambahkan baris berikut sebelum `pool` pernyataan `server` atau pernyataan lain yang mungkin ada dalam file, dan simpan perubahan Anda:

   ```
   server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
   ```

4. Mulai ulang daemon `chrony` (`chronyd`).

   ```
   [ec2-user ~]$ sudo service chronyd restart
   ```

   ```
   Starting chronyd:                                          [  OK  ]
   ```

> **Note**
>
> Di RHEL dan CentOS (hingga versi 6), nama layanannya adalah `chrony`, bukan `chronyd`.

5. Untuk mengonfigurasi `chronyd` agar dimulai di setiap boot sistem, gunakan perintah `chkconfig`.

```
[ec2-user ~]$ sudo chkconfig chronyd on
```

6. Pastikan bahwa `chrony` menggunakan titik akhir IPv4 `169.254.169.123` untuk menyinkronkan waktu.

```
[ec2-user ~]$ chronyc sources -v | grep -F ^*
```

Dalam output, `^*` menunjukkan sumber waktu yang disukai.

```
^* 169.254.169.123               3   6    17    43    -30us[ -226us] +/-  287us
```

7. Verifikasi metrik sinkronisasi waktu yang dilaporkan oleh `chrony`.

```
[ec2-user ~]$ chronyc tracking
```

```
Reference ID    : A9FEA97B (169.254.169.123)
Stratum         : 4
Ref time (UTC)  : Wed May 06 00:39:14 2026
System time     : 0.000002191 seconds fast of NTP time
Last offset     : +0.000002164 seconds
RMS offset      : 0.000082968 seconds
Frequency       : 3.710 ppm slow
Residual freq   : +0.002 ppm
Skew            : 0.504 ppm
Root delay      : 0.000362541 seconds
Root dispersion : 0.000225028 seconds
Update interval : 16.1 seconds
Leap status     : Normal
```

### Ubuntu

Untuk mengkonfigurasi chrony untuk terhubung ke titik akhir IPv4 di Ubuntu

1. Hubungkan ke instans Anda dan gunakan `apt` untuk memasang paket `chrony`.

```
ubuntu:~$ sudo apt install chrony
```

> **Note**
>
> Jika perlu, perbarui instans Anda terlebih dahulu dengan menjalankan `sudo apt update`.

2. Buka file `/etc/chrony/chrony.conf` menggunakan editor teks (seperti vim atau nano). Tambahkan baris sebelum `server` yang lain atau pernyataan `pool` yang sudah ada di file, dan simpan perubahan Anda.

```
server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
```

3. Mulai ulang layanan `chrony`.

```
ubuntu:~$ sudo /etc/init.d/chrony restart
```

```
Restarting chrony (via systemctl): chrony.service.
```

4. Pastikan bahwa `chrony` menggunakan titik akhir IPv4 `169.254.169.123` untuk menyinkronkan waktu.

```
ubuntu:~$ chronyc sources -v | grep -F ^*
```

Dalam output, baris yang dimulai dengan `^*` menunjukkan sumber waktu yang disukai.

```
^* 169.254.169.123               3   6    17    12    +15us[  +57us] +/-  320us
```

5. Verifikasi metrik sinkronisasi waktu yang dilaporkan oleh `chrony`.

```
ubuntu:~$ chronyc tracking
```

```
Reference ID    : A9FEA97B (169.254.169.123)
Stratum         : 4
Ref time (UTC)  : Wed May 06 00:39:14 2026
System time     : 0.000002191 seconds fast of NTP time
Last offset     : +0.000002164 seconds
RMS offset      : 0.000082968 seconds
Frequency       : 3.710 ppm slow
Residual freq   : +0.002 ppm
Skew            : 0.504 ppm
Root delay      : 0.000362541 seconds
Root dispersion : 0.000225028 seconds
Update interval : 16.1 seconds
Leap status     : Normal
```

SUSE Linux

Dimulai dengan SUSE Linux Enterprise Server 15, `chrony` adalah implementasi default NTP.

Untuk mengkonfigurasi chrony untuk terhubung ke titik akhir IPv4 di SUSE Linux

1. Buka file `/etc/chrony.conf` menggunakan editor teks (seperti vim atau nano).
2. Pastikan bahwa file tersebut berisi baris berikut:

   ```
   server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
   ```

   Jika baris ini tidak ada, tambahkan.
3. Komentari baris server atau kolam lainnya.
4. Buka YaST dan aktifkan layanan chrony.

Windows

Sejak rilis Agustus 2018, AMI Windows menggunakan Layanan Amazon Time Sync secara default. Konfigurasi lebih lanjut tidak diperlukan untuk instans yang diluncurkan dari AMI ini dan Anda dapat melewatkan prosedur berikut.

Jika Anda menggunakan AMI yang tidak memiliki Amazon Time Sync Service yang dikonfigurasi secara default, verifikasi terlebih dahulu konfigurasi NTP Anda saat ini. Jika instans Anda sudah menggunakan titik akhir IPv4 dari Layanan Amazon Time Sync, konfigurasi lebih lanjut tidak

diperlukan. Jika instans Anda tidak menggunakan Layanan Amazon Time Sync, selesaikan prosedur untuk mengubah server NTP agar menggunakan Layanan Amazon Time Sync.

Untuk memverifikasi konfigurasi NTP

1. Dari instans Anda, buka jendela Command Prompt.
2. Dapatkan konfigurasi NTP saat ini dengan mengetikkan perintah berikut:

   ```
   w32tm /query /configuration
   ```

   Perintah ini mengembalikan pengaturan konfigurasi saat ini untuk instans Windows dan akan ditampilkan jika Anda terhubung ke Layanan Amazon Time Sync.
3. (Opsional) Dapatkan status konfigurasi saat ini dengan mengetik perintah berikut:

   ```
   w32tm /query /status
   ```

   Perintah ini mengembalikan informasi seperti terakhir kali instans disinkronkan dengan server NTP dan interval polling.

Untuk mengubah server NTP agar menggunakan Layanan Amazon Time Sync

1. Dari jendela Command Prompt, jalankan perintah berikut:

   ```
   w32tm /config /manualpeerlist:169.254.169.123 /syncfromflags:manual /update
   ```
2. Verifikasi pengaturan baru Anda dengan menggunakan perintah berikut:

   ```
   w32tm /query /configuration
   ```

   Dalam output yang dikembalikan, pastikan bahwa `NtpServer` menampilkan titik akhir IPv4 `169.254.169.123`.

Pengaturan NTP default untuk AMI Amazon Windows

Amazon Machine Image (AMI) umumnya mematuhi pengaturan bawaan siap pakai kecuali jika perubahan diperlukan untuk berfungsi pada infrastruktur EC2. Pengaturan berikut telah ditetapkan

agar bekerja dengan baik di lingkungan virtual, serta untuk menjaga agar setiap perbedaan waktu tetap dalam akurasi satu detik:

- Interval  Pembaruan - Mengatur seberapa sering layanan waktu akan menyesuaikan waktu sistem menuju akurasi. AWS mengkonfigurasi interval pembaruan untuk terjadi setiap dua menit sekali.
- Server NTP – Sejak rilis Agustus 2018, AMI menggunakan Layanan Amazon Time Sync secara default. Layanan waktu ini dapat diakses dari mana saja Wilayah AWS di titik akhir IPv4 169.254.169.123. Selain itu, bendera 0x9 menunjukkan bahwa layanan waktu bertindak sebagai klien, dan untuk menggunakan `SpecialPollInterval` untuk menentukan seberapa sering pemeriksaan dengan server waktu yang dikonfigurasi.
- Tipe - "NTP" berarti bahwa layanan bertindak sebagai klien NTP mandiri, alih-alih bertindak sebagai bagian dari domain.
- Diaktifkan dan InputProvider  - Layanan waktu diaktifkan dan memberikan waktu ke sistem operasi.
- Interval Polling Khus  us - Memeriksa terhadap Server NTP yang dikonfigurasi setiap 900 detik (15 menit).

> **Note**
>
> Untuk AMI Windows Server 2025, nil `SpecialPollInterval` ainya adalah 1024 detik, bukan 900 detik.

| Jalur registri | Nama kunci | Data |
| --- | --- | --- |
| HKL M:\System\\ layananCurrentControlSet\ w32time\ Konfigurasi | UpdateInterval | 120 |
| HKL M:\System\\ layananCurrentControlSet\ w32time\ Parameter | NtpServer | 169.254.169.123,0x9 |

| Jalur registri | Nama kunci | Data |
| --- | --- | --- |
| HKL M:\System\\ layananCurrentControlSet\ w32time\ Parameter | Tipe | NTP |
| HKL M:\System\\ layananCurrentControlSet\ w32waktu\\ TimeProviders NtpClient | Aktif | 1 |
| HKL M:\System\\ layananCurrentControlSet\ w32waktu\\ TimeProviders NtpClient | InputProvider | 1 |
| HKL M:\System\\ layananCurrentControlSet\ w32waktu\\ TimeProviders NtpClient | SpecialPollInterval | 900 (Windows Server 2016, 2019, dan 2022) atau 1024 (Windows Server 2025) |

## Akses titik akhir IPv6 dari Layanan Sinkronisasi Waktu Amazon

Bagian ini menjelaskan perbedaan langkah-langkah yang dijelaskan dalam Akses titik akhir IPv4 dari Layanan Sinkronisasi Waktu Amazon jika Anda mengonfigurasi instans untuk menggunakan Layanan Amazon Time Sync lokal melalui titik akhir IPv6. Bagian ini tidak menjelaskan seluruh proses konfigurasi Layanan Amazon Time Sync.

Titik akhir IPv6 hanya dapat diakses pada Nitro-based instance.

Kami tidak menyarankan menggunakan entri titik akhir IPv4 dan IPv6 secara bersamaan. Paket NTP IPv4 dan IPv6 berasal dari server lokal yang sama untuk instans Anda. Konfigurasi titik akhir IPv4 dan IPv6 tidak diperlukan dan tidak akan meningkatkan akurasi waktu pada instans Anda.

Linux

Bergantung pada distribusi Linux yang Anda gunakan, ketika Anda mencapai langkah untuk mengedit `chrony.conf` file, Anda akan menggunakan titik akhir IPv6 dari Amazon Time Sync Service (`fd00:ec2::123`) daripada titik akhir IPv4 (): `169.254.169.123`

```
server fd00:ec2::123 prefer iburst minpoll 4 maxpoll 4
```

Simpan file dan verifikasi bahwa chrony menggunakan titik akhir `fd00:ec2::123` IPv6 untuk menyinkronkan waktu:

```
[ec2-user ~]$ chronyc sources -v
```

Dalam output, jika Anda melihat titik akhir `fd00:ec2::123` IPv6, konfigurasi selesai.

Windows

Ketika Anda mencapai langkah untuk mengubah server NTP untuk menggunakan Amazon Time Sync Service, Anda akan menggunakan titik akhir IPv6 dari Amazon Time Sync Service (`fd00:ec2::123`) daripada titik akhir IPv4 (): `169.254.169.123`

```
w32tm /config /manualpeerlist:fd00:ec2::123 /syncfromflags:manual /update
```

Pastikan pengaturan baru Anda menggunakan titik akhir `fd00:ec2::123` IPv6 untuk menyinkronkan waktu:

```
w32tm /query /configuration
```

Dalam output, verifikasi bahwa `NtpServer` menampilkan titik akhir `fd00:ec2::123` IPv6.

## Akses Layanan Sinkronisasi Waktu Amazon yang disempurnakan

Layanan Sinkronisasi Waktu Amazon yang disempurnakan menawarkan sumber waktu lokal akurasi yang lebih tinggi untuk instans Amazon EC2 yang didukung. Instans yang diluncurkan dalam grup penempatan dengan `precision-time` strategi dapat mengakses sumber lokal ini. Kami merekomendasikan grup penempatan waktu presisi untuk aplikasi yang memerlukan waktu yang lebih akurat baik dari sumber NTP link-local, atau perangkat Jam Perangkat Keras PTP (PHC) pada instance Linux. Saat Anda meluncurkan instans ke grup penempatan waktu presisi, AWS tempatkan instans pada perangkat keras yang didukung dengan akses langsung ke sumber waktu presisi tinggi di AWS infrastruktur.

Manfaat utama

- Sumber NTP yang ditingkatkan secara default - Instans Anda memiliki akses langsung ke sumber waktu NTP lokal yang ditingkatkan, jika Anda menggunakan alamat IP link-lokal NTP seperti yang dijelaskan di bagian sebelumnya.

- Microsecond-accurate sinkronisasi jam — Konfigurasikan instans Amazon EC2 Linux Anda untuk menggunakan perangkat Jam Perangkat Keras PTP dan mencapai sinkronisasi jam yang akurat mikrodetik.
- Penerapan yang disederhanakan — Strategi penempatan tunggal memastikan semua instans memiliki kemampuan waktu yang presisi.
- Timestamp paket perangkat keras — Akses stempel waktu paket tingkat rendah untuk pengukuran jaringan.
- Tidak ada biaya tambahan - Grup penempatan waktu presisi tersedia tanpa biaya tambahan.

Aturan dan batasan

- Grup penempatan waktu presisi tersedia di semua Wilayah AWS Komersil.
- Grup penempatan waktu presisi mendukung rangkaian instans Amazon EC2 Gen7 dan versi lebih baru berikut:
  - Tujuan umum: M7a, M7g, M7gd, M7i,, M8a, M8g M7i-flex
  - Komputasi dioptimalkan: C7a, C7gd, C7i,, C8g, C8gd C7i-flex
  - Memori dioptimalkan: R7a, R7g, R7i, R8g, X8g
  - Penyimpanan dioptimalkan: i8G, i8GE
- Jika Anda memulai atau meluncurkan instance dalam grup penempatan waktu presisi dan perangkat keras tidak cukup untuk memberikan akses ke Layanan Sinkronisasi Waktu Amazon yang disempurnakan, permintaan gagal.
- Jika Anda menghentikan instance dalam grup penempatan waktu presisi dan kemudian memulainya lagi, instans masih berjalan di grup penempatan. Namun, awal mungkin gagal jika perangkat keras tidak mencukupi untuk memberikan akses ke Layanan Sinkronisasi Waktu Amazon yang ditingkatkan.
- Amazon EC2 terus menambahkan perangkat keras yang mendukung Layanan Sinkronisasi Waktu Amazon yang ditingkatkan. Jika permintaan Anda gagal karena kapasitas yang tidak mencukupi, coba lagi nanti atau coba Zona Ketersediaan yang berbeda. Untuk informasi selengkapnya, lihat Memecahkan masalah peluncuran instans Amazon EC2.
- Aturan dan batasan kelompok penempatan berlaku. Untuk informasi selengkapnya, lihat Grup penempatan untuk instans Amazon EC2 Anda

### Buat grup penempatan waktu presisi

Anda dapat membuat grup penempatan waktu presisi menggunakan AWS CLI, AWS Management Console, atau AWS SDK, dengan menentukan `precision-time` strategi.

#### Penggunaan AWS CLI

Jalankan perintah berikut:

```
aws ec2 create-placement-group \
    --group-name my-precision-time-pg \
    --strategy precision-time
```

Perintah mengembalikan grup penempatan ARN yang Anda gunakan saat membuat reservasi kapasitas, dan nama grup atau ID grup yang Anda gunakan saat meluncurkan instance dan menautkan grup penempatan.

### Luncurkan sebuah instans

Setelah Anda membuat grup penempatan waktu presisi, tentukan grup tersebut saat meluncurkan instans untuk mengakses Layanan Sinkronisasi Waktu Amazon yang disempurnakan:

```
aws ec2 run-instances \
    --image-id ami-0abcdef1234567890 \
    --instance-type r7g.2xlarge \
    --placement GroupId=pg-0aaa1111111111111
```

### Verifikasi akses ke Layanan Sinkronisasi Waktu Amazon yang disempurnakan

Setelah meluncurkan instans dalam grup penempatan waktu presisi, Anda mendapat manfaat dari sumber waktu NTP yang disempurnakan.

Misalnya, jika Anda menggunakan daemon chronyd pada instance Anda, Anda dapat memverifikasi bahwa sumber waktu NTP sekarang disebut sebagai Stratum 1 dan telah meningkatkan metrik akurasi jam:

```
[ec2-user ~]$ chronyc sources
```

```
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^* 169.254.169.123               1   4   377     3  +3477ns[+4689ns] +/-   91us
```

Verifikasi metrik sinkronisasi waktu yang dilaporkan oleh `chrony`.

```
[ec2-user ~]$ chronyc tracking
```

```
Reference ID    : A9FEA97B (169.254.169.123)
Stratum         : 2
Ref time (UTC)  : Wed May 06 01:33:43 2026
System time     : 0.000000276 seconds fast of NTP time
Last offset     : +0.000000331 seconds
RMS offset      : 0.000001929 seconds
Frequency       : 2.870 ppm fast
Residual freq   : +0.000 ppm
Skew            : 0.031 ppm
Root delay      : 0.000107584 seconds
Root dispersion : 0.000036476 seconds
Update interval : 16.2 seconds
Leap status     : Normal
```

## Akses Jam Perangkat Keras PTP (PHC)

PTP Hardware Clock (PHC) adalah bagian dari Sistem AWS Nitro. Ini dapat diakses langsung pada instans bare metal yang didukung dan instans Amazon EC2 virtual yang diluncurkan dalam grup penempatan waktu presisi. Perangkat PHC saat ini hanya dapat diakses pada instans Linux. Bagian berikut menjelaskan cara mengatur dan memverifikasi perangkat PHC pada instance Linux Anda.

### Persyaratan

- Driver ENA versi 2.10.0 atau setelahnya diinstal pada sistem operasi yang didukung.
- Instans Amazon EC2 yang menjalankan Linux. Untuk informasi selengkapnya tentang sistem operasi yang didukung, lihat prasyarat driver di GitHub.
- Instans Amazon EC2 diluncurkan dalam grup penempatan waktu presisi (lihatAkses Layanan Sinkronisasi Waktu Amazon yang disempurnakan).

> **Note**
>
> Layanan Sinkronisasi Waktu Amazon yang disempurnakan dan perangkat PHC tetap dapat diakses tanpa grup penempatan waktu presisi di wilayah berikut dan untuk keluarga instans tertentu. Untuk pengalaman terbaik, sebaiknya luncurkan instans Amazon EC2 dalam grup penempatan presisi sebagai gantinya.

- Akses lama didukung Wilayah AWS: AS Timur (Virginia Utara), AS Timur (Ohio), Asia Pasifik (Malaysia), Asia Pasifik (Thailand), Asia Pasifik (Tokyo), dan Eropa (Stockholm)
- Akses lama didukung Zona Lokal: AS Timur (Kota New York)
- Keluarga instans yang didukung akses lama:
  - Tujuan umum: M7a, M7g, M7i
  - Memori dioptimalkan: R7a, R7g, R7i
  - Penyimpanan dioptimalkan: i8G, i8GE

Kompilasi dan aktifkan driver ENA dengan dukungan PHC

Selalu tinjau instruksi terbaru dari versi driver Elastic Network Adapter (ENA) terbaru dalam dokumentasi driver https://github.com/amzn/amzn-drivers/tree/master/kernel/linux/ena#ptp-hardware-clock-phc ENA GitHub untuk informasi khusus untuk sistem operasi Anda. Berikut ini adalah ikhtisar langkah-langkah untuk Amazon Linux.

Sebelum Anda mulai, pastikan instans Anda memenuhi persyaratanPersyaratan.

Untuk mengkompilasi dan mengaktifkan driver ENA dengan dukungan PHC

1. Instal prasyarat.

```
[ec2-user ~]$ sudo yum update
[ec2-user ~]$ sudo yum install kernel-devel-$(uname -r) git
[ec2-user ~]$ sudo reboot
```

2. Muat modul ptp yang diperlukan.

```
[ec2-user ~]$ sudo modprobe ptp
```

3. Ambil versi driver ENA terbaru (versi 2.10.0 atau yang lebih baru).

```
[ec2-user ~]$ git clone https://github.com/amzn/amzn-drivers.git /tmp/amzn-drivers
```

4. Bangun driver ENA `ena.ko` dengan dukungan PHC.

```
[ec2-user ~]$ cd /tmp/amzn-drivers/kernel/linux/ena
[ec2-user ~]$ ENA_PHC_INCLUDE=1 make
```

5. Muat ulang driver ENA dan aktifkan perangkat PHC.

```
[ec2-user ~]$ sudo rmmod ena && sudo insmod ena.ko phc_enable=1
```

6. Validasi driver ENA yang dimuat memiliki dukungan PHC.

```
[ec2-user ~]$ modinfo ena | grep -E "phc_enable"
```

```
parm:           phc_enable:Enable PHC.
```

7. Validasi dukungan PHC diaktifkan dengan driver ENA.

```
[ec2-user ~]$ cat /sys/module/ena/parameters/phc_enable
```

```
1
```

Untuk petunjuk untuk menginstal driver dan mengaktifkan opsi phc_enable saat reboot, lihat READ ME driver ENA aktif. GitHub

Verifikasi konfigurasi perangkat PTP

Pastikan perangkat jam perangkat keras ENA PTP muncul di instans Anda.

```
[ec2-user ~]$ for file in /sys/class/ptp/*; do echo -n "$file: "; cat "$file/
clock_name"; done
```

Output yang diharapkan

```
/sys/class/ptp/ptp<index>: ena-ptp-<PCI slot>
```

Di mana:

- *index* adalah indeks jam perangkat keras PTP yang terdaftar di kernel.
- *PCI slot* adalah slot PCI pengontrol ethernet ENA. Ini adalah slot yang sama seperti yang ditunjukkan pada `lspci | grep ENA`.

Contoh Output

```
/sys/class/ptp/ptp0: ena-ptp-05
```

Jika `ena-ptp-<PCI slot>` tidak ada dalam output, driver ENA tidak akan diinstal dengan benar. Tinjau langkah-langkahnyaKompilasi dan aktifkan driver ENA dengan dukungan PHC.

Konfigurasikan symlink PTP

Perangkat PTP biasanya diberi nama`/dev/ptp0`,`/dev/ptp1`, dan seterusnya, dengan indeksnya tergantung pada urutan inisialisasi perangkat keras. Membuat symlink memastikan bahwa aplikasi seperti chrony secara konsisten mereferensikan perangkat yang benar, terlepas dari perubahan indeks.

AMI Amazon Linux 2023 terbaru menyertakan `udev` aturan yang membuat `/dev/ptp_ena` symlink, menunjuk ke `/dev/ptp` entri yang benar terkait dengan host ENA.

Pertama, periksa apakah symlink hadir dengan menjalankan perintah berikut.

```
[ec2-user ~]$ ls -l /dev/ptp*
```

Contoh Output

```
crw------- 1 root root 245, 0 Jan 31 2025 /dev/ptp0
lrwxrwxrwx 1 root root      4 Jan 31 2025 /dev/ptp_ena -> ptp0
```

Di mana:

- `/dev/ptp<index>`adalah jalur ke perangkat PTP.
- `/dev/ptp_ena`adalah symlink konstan, yang menunjuk ke perangkat PTP yang sama.

Jika `/dev/ptp_ena` symlink ada, lewati keKonfigurasikan daemon chronyd untuk menggunakan perangkat PHC. Jika hilang, lakukan hal berikut:

Untuk membuat symlink PTP

1. Tambahkan `udev` aturan berikut.

   ```
   [ec2-user ~]$ echo "SUBSYSTEM==\"ptp\", ATTR{clock_name}==\"ena-ptp-*\", SYMLINK +=
    \"ptp_ena\"" | sudo tee -a /etc/udev/rules.d/53-ec2-network-interfaces.rules
   ```

2. Muat ulang `udev` aturan, baik dengan me-reboot instance, atau dengan menjalankan perintah berikut.

```
[ec2-user ~]$ sudo udevadm control --reload-rules && sudo udevadm trigger
```

Konfigurasikan daemon chronyd untuk menggunakan perangkat PHC

Daemon sinkronisasi jam chronyd harus dikonfigurasi untuk menggunakan perangkat PHC sebagai sumber waktu tambahan, menggunakan `/dev/ptp_ena` symlink untuk mengidentifikasi perangkat.

Untuk mengkonfigurasi daemon chronyd untuk menggunakan perangkat PHC

1. Edit `/etc/chrony.conf` menggunakan editor teks dan tambahkan baris berikut:

```
refclock PHC /dev/ptp_ena poll 0 delay 0.000010 prefer
```

2. Mulai ulang chrony.

```
[ec2-user ~]$ sudo systemctl restart chronyd
```

3. Verifikasi bahwa chrony menggunakan jam perangkat keras PTP. Perangkat PHC0 harus menjadi sumber waktu yang disukai dengan nilai Stratum 0. Sumber waktu NTP lokal (jika dikonfigurasi) harus memiliki nilai Stratum 1.

```
[ec2-user ~]$ chronyc sources
```

```
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
#* PHC0                          0   0   377     0   +184ns[ +198ns] +/- 5032ns
^- 169.254.169.123             1   4   377     8    -18us[  -18us] +/-  115us
```

Verifikasi stempel waktu paket perangkat keras

Driver ENA yang dimuat dengan dukungan PHC menyediakan akses ke timestamp paket perangkat keras. Anda dapat memverifikasi dukungan untuk fungsi ini dengan menggunakan `ethtool -T` *`interface`* perintah.

Untuk informasi selengkapnya tentang menggunakan stempel waktu paket perangkat keras, lihat dokumentasi paket timestamp Linux.

```
[ec2-user ~]$ sudo ethtool -T ens5
```

```
Time stamping parameters for ens5:
Capabilities:
        software-transmit
        hardware-receive
        software-receive
        software-system-clock
PTP Hardware Clock: 0
Hardware Transmit Timestamp Modes: none
Hardware Receive Filter Modes:
        none
        all
```

Jika output ditampilkan `hardware-receive` dalam daftar Capabilities, timestamp paket perangkat keras tersedia di instans Anda.

## Mengatur referensi waktu pada instans EC2 Anda atau perangkat yang terhubung ke internet untuk menggunakan Layanan Sinkronisasi Waktu Amazon publik

Anda dapat mengatur instans Anda, atau perangkat apa pun yang terhubung ke internet seperti komputer lokal atau server on-premis, agar menggunakan Layanan Amazon Time Sync publik, yang dapat diakses melalui internet di `time.aws.com`. Anda dapat menggunakan Layanan Sinkronisasi Waktu Amazon publik sebagai cadangan untuk Layanan Sinkronisasi Waktu Amazon lokal dan untuk menghubungkan sumber daya di luar AWS ke Layanan Sinkronisasi Waktu Amazon.

> **Note**
>
> Untuk performa terbaik, sebaiknya gunakan Layanan Sinkronisasi Waktu Amazon lokal pada instans Anda, dan hanya menggunakan Layanan Sinkronisasi Waktu Amazon publik sebagai cadangan.

Gunakan instruksi untuk sistem operasi instans atau perangkat Anda.

Linux

Untuk mengatur instans atau perangkat Linux Anda agar menggunakan Layanan Amazon Time Sync publik menggunakan chrony atau ntpd

1. Edit `/etc/chrony.conf` (jika Anda menggunakan chrony) atau `/etc/ntp.conf` (jika Anda menggunakan ntpd) menggunakan editor teks sebagai berikut:

   a. Untuk mencegah instans atau perangkat Anda mencoba mencampur server yang dioleskan dan yang tidak diolesi, hapus atau komentari baris yang dimulai `server` kecuali koneksi yang ada ke Layanan Sinkronisasi Waktu Amazon lokal.

   > **Important**
   >
   > Jika Anda menyetel instans EC2 agar terhubung ke Layanan Amazon Time Sync publik, jangan hapus baris berikut yang menetapkan instans yang akan dihubungkan ke Layanan Amazon Time Sync. Layanan Amazon Time Sync lokal adalah koneksi yang lebih langsung dan akan memberikan akurasi jam yang lebih baik. Layanan Amazon Time Sync publik hanya boleh digunakan sebagai cadangan.
   >
   > ```
   > server 169.254.169.123 prefer iburst minpoll 4 maxpoll 4
   > ```

   b. Tambahkan baris berikut untuk terhubung ke Layanan Amazon Time Sync publik.

   ```
   pool time.aws.com iburst
   ```

2. Mulai ulang daemon menggunakan salah satu perintah berikut.

   - chrony

     ```
     sudo service chronyd force-reload
     ```

   - ntpd

     ```
     sudo service ntp reload
     ```

macOS

Untuk mengatur instans atau perangkat macOS Anda agar menggunakan Layanan Amazon Time Sync publik

1. Buka Preferensi Sistem.
2. Pilih Tanggal & Waktu, lalu pilih tab Tanggal & Waktu.
3. Untuk melakukan perubahan, pilih ikon kunci, dan masukkan kata sandi Anda saat diminta.
4. Untuk Atur tanggal dan waktu secara otomatis, masukkan **`time.aws.com`**.

Windows

Untuk mengatur instans atau perangkat Windows Anda agar menggunakan Layanan Amazon Time Sync publik

1. Buka Panel Kontrol.
2. Pilih ikon Tanggal dan Waktu.
3. Pilih tab Waktu Internet. Tab ini tidak tersedia jika PC Anda adalah bagian dari domain. Dalam hal ini, waktu akan disinkronkan dengan pengontrol domain. Anda dapat mengonfigurasi pengontrol untuk menggunakan Amazon Time Sync Service publik.
4. Pilih Ubah pengaturan.
5. Pilih kotak centang untuk Sin  kronisasi dengan server  waktu Internet.
6. Di sebelah Server, masukkan **`time.aws.com`**.

Untuk mengatur instans atau perangkat Windows Server Anda agar menggunakan Layanan Amazon Time Sync publik

- Ikuti Instruksi Microsoft untuk memperbarui registri Anda.

## Bandingkan stempel waktu untuk instans Linux Anda

Jika Anda menggunakan Layanan Sinkronisasi Waktu Amazon, Anda dapat membandingkan cap waktu pada instans Amazon EC2 Linux Anda dengan ClockBound untuk menentukan waktu sebenarnya dari suatu peristiwa. ClockBound mengukur akurasi jam instans EC2 Anda, dan memungkinkan Anda untuk memeriksa apakah stempel waktu tertentu berada di masa lalu atau

masa depan sehubungan dengan jam instans Anda saat ini. Informasi ini berguna untuk menentukan urutan dan konsistensi peristiwa dan transaksi di seluruh instans EC2, terlepas dari lokasi geografis masing-masing instans.

ClockBound adalah daemon dan perpustakaan open source. Untuk mempelajari lebih lanjut tentang ClockBound, termasuk petunjuk pemasangan, lihat ClockBound di GitHub.

ClockBound hanya didukung untuk instance Linux.

Jika Anda menggunakan koneksi PTP langsung ke jam perangkat keras PTP, daemon waktu Anda, seperti chrony, akan meremehkan batas kesalahan jam. Hal ini karena jam perangkat keras PTP tidak meneruskan informasi batas kesalahan yang benar ke chrony, seperti yang dilakukan NTP. Akibatnya, daemon sinkronisasi jam Anda mengasumsikan jam akurat hingga ke UTC dan dengan demikian memiliki batas kesalahan `0`. Untuk mengukur batas kesalahan penuh, Sistem Nitro menghitung batas kesalahan jam perangkat keras PTP, dan membuatnya tersedia untuk instans EC2 Anda melalui sistem file driver ENA. `sysfs` Anda dapat membaca ini secara langsung sebagai nilai, dalam nanodetik.

Untuk mengambil terikat kesalahan jam perangkat keras PTP

1. Pertama dapatkan lokasi yang benar dari perangkat jam perangkat keras PTP dengan menggunakan salah satu perintah berikut. Jalur dalam perintah berbeda tergantung pada AMI yang digunakan untuk meluncurkan instance.

   - Untuk Amazon Linux 2:

     ```
     cat /sys/class/net/eth0/device/uevent | grep PCI_SLOT_NAME
     ```

   - Untuk Amazon Linux 2023:

     ```
     cat /sys/class/net/ens5/device/uevent | grep PCI_SLOT_NAME
     ```

   Outputnya adalah nama slot PCI, yang merupakan lokasi perangkat jam perangkat keras PTP. Dalam contoh ini, lokasinya adalah`0000:00:03.0`.

   ```
   PCI_SLOT_NAME=0000:00:03.0
   ```

2. Untuk mengambil terikat kesalahan jam perangkat keras PTP, jalankan perintah berikut. Sertakan nama slot PCI dari langkah sebelumnya.

```
cat /sys/bus/pci/devices/0000:00:03.0/phc_error_bound
```

Output-nya adalah batas kesalahan jam pada jam perangkat keras PTP, dalam nanodetik.

Untuk menghitung batas kesalahan jam yang benar pada titik waktu tertentu saat menggunakan koneksi PTP langsung ke jam perangkat keras PTP, Anda harus menambahkan batas kesalahan jam dari chrony atau ClockBound pada waktu yang mensurvei jam perangkat chrony keras PTP. Untuk informasi selengkapnya tentang mengukur dan memantau akurasi jam, lihat Mengelola akurasi jam instans Amazon EC2 menggunakan Amazon Time Sync Service dan Amazon CloudWatch — Bagian 1.

## Mengubah zona waktu instans Anda

Instans Amazon EC2 diatur ke zona waktu UTC (Coordinated Universal Time) secara default. Anda dapat mengubah waktu pada sebuah instans ke zona waktu lokal atau ke zona waktu lain di jaringan Anda.

Gunakan instruksi untuk sistem operasi instans Anda.

Linux

> **Important**
>
> Informasi ini berlaku untuk Amazon Linux. Untuk informasi tentang distribusi lain, lihat dokumentasi spesifik tentangnya.

Untuk mengubah zona waktu di Amazon Linux

1. Lihat pengaturan zona waktu sistem saat ini.

   ```
   [ec2-user ~]$ timedatectl
   ```

2. Buat daftar zona waktu yang tersedia.

   ```
   [ec2-user ~]$ timedatectl list-timezones
   ```

3. Atur zona waktu yang dipilih.

```
[ec2-user ~]$ sudo timedatectl set-timezone America/Vancouver
```

4. (Opsional) Konfirmasikan bahwa zona waktu saat ini diperbarui ke zona waktu baru dengan menjalankan perintah timedatectl lagi.

```
[ec2-user ~]$ timedatectl
```

Windows

Untuk mengubah zona waktu pada instans Windows

1. Dari instans Anda, buka jendela Command Prompt.
2. Identifikasi zona waktu yang akan digunakan pada instans. Untuk mendapatkan daftar zona waktu, gunakan perintah berikut:

```
tzutil /l
```

Perintah ini mengembalikan daftar semua zona waktu yang tersedia dalam format berikut:

```
display name
time zone ID
```

3. Temukan ID zona waktu untuk ditetapkan ke instans.
4. Contoh: Tetapkan zona waktu UTC:

```
tzutil /s "UTC"
```

Contoh: Tetapkan Waktu Standar Pasifik:

```
tzutil /s "Pacific Standard Time"
```

Saat Anda mengubah zona waktu pada instans Windows, Anda harus memastikan bahwa zona waktu tidak berubah hingga sistem dimulai ulang. Jika tidak, saat dimulai ulang, instans akan kembali menggunakan waktu UTC. Anda dapat mempertahankan pengaturan zona waktu Anda dengan menambahkan kunci RealTimeIsUniversal registri. Kunci ini disetel secara default pada semua

instans generasi saat ini. Untuk memverifikasi apakah kunci RealTimeIsUniversal registri disetel, lihat langkah 3 dalam prosedur berikut. Jika kuncinya belum diatur, ikuti langkah-langkah ini dari awal.

Untuk mengatur kunci RealTimeIsUniversal registri

1. Dari instans Anda, buka jendela Command Prompt.
2. Gunakan perintah berikut untuk menambahkan kunci registri:

```
reg add "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation" /
v RealTimeIsUniversal /d 1 /t REG_DWORD /f
```

3. (Opsional) Pastikan instans tersebut berhasil menyimpan kunci menggunakan perintah berikut:

```
reg query "HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control
\TimeZoneInformation" /s
```

   Perintah ini mengembalikan subkunci untuk kunci registri TimeZoneInformation. Anda harus melihat kunci RealTimeIsUniversal di bagian bawah daftar, mirip dengan yang berikut ini:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TimeZoneInformation
    Bias                            REG_DWORD     0x1e0
    DaylightBias                    REG_DWORD     0xffffffc4
    DaylightName                    REG_SZ        @tzres.dll,-211
    DaylightStart                   REG_BINARY    00000300020002000000000000000000
    StandardBias                    REG_DWORD     0x0
    StandardName                    REG_SZ        @tzres.dll,-212
    StandardStart                   REG_BINARY    00000B0001000200000000000000000
    TimeZoneKeyName                 REG_SZ        Pacific Standard Time
    DynamicDaylightTimeDisabled     REG_DWORD     0x0
    ActiveTimeBias                  REG_DWORD     0x1a4
    RealTimeIsUniversal             REG_DWORD     0x1
```

# Manajer Kapasitas EC2

Amazon EC2 Capacity Manager membantu Anda memantau, menganalisis, dan mengelola penggunaan kapasitas di seluruh On-Demand Instans, Instans Spot, dan Reservasi Kapasitas dari satu lokasi. Capacity Manager menyederhanakan pelacakan kapasitas melalui dasbor terpadu yang menggabungkan data penggunaan Anda dengan penyegaran setiap jam dan peluang pengoptimalan.

Dirancang untuk perusahaan besar dan organisasi multi-akun, Capacity Manager menghilangkan pengumpulan data manual dari berbagai sumber. Alat ini memberikan wawasan mendalam di seluruh Anda Wilayah AWS. Anda dapat menelusuri metrik kapasitas terperinci dan mengambil tindakan segera untuk mengoptimalkan sumber daya cloud Anda.

Saat Anda mengaktifkan integrasi AWS Organisasi, Anda dapat melihat dan menganalisis data kapasitas di semua akun anggota di organisasi Anda dari satu dasbor. Tanpa integrasi Organisasi, Manajer Kapasitas hanya memantau sumber daya dalam Akun AWS individu yang diaktifkan.

> **Note**
>
> - Anda hanya dapat mengaktifkan Manajer Kapasitas dalam satu Wilayah AWS per akun.
> - Semua biaya yang ditampilkan di Manajer Kapasitas adalah perkiraan biaya berdasarkan On-Demand harga yang dipublikasikan. Perkiraan ini tidak termasuk diskon seperti Paket Tabungan atau Instans Cadangan. Biaya aktual Anda mungkin berbeda dari perkiraan ini.

## Fitur Utama

- Dasbor — Menyediakan ikhtisar tingkat tinggi dari semua Reservasi On-Demand Kapasitas, On-Demand dan penggunaan Spot, termasuk metrik utama untuk membantu meningkatkan postur kapasitas Anda.
- Cross-account dan agregasi lintas wilayah — Lihat penggunaan kapasitas di semua akun anggota di organisasi Anda dari satu dasbor. Capacity Manager mengumpulkan data dari semua Wilayah dalam partisi ke dalam satu Wilayah tempat Anda telah mengaktifkan Manajer Kapasitas, termasuk data tag. Untuk data lintas akun, Manajer Kapasitas juga menyediakan nama akun sebagai dimensi.
- Tag-based pengelompokan dan memfilter — Aktifkan kunci tag dari sumber daya Amazon EC2 Anda (misalnya, `environment` atau`team`) untuk digunakan sebagai dimensi saat mengelompokkan dan memfilter metrik kapasitas Anda. Manager-providedTag kapasitas untuk Grup Penskalaan Otomatis EC2, nama cluster EKS, kumpulan simpul EKS Kubernetes, dan kumpulan node Karpenter disertakan secara default. Untuk informasi selengkapnya, lihat Mengelola kunci tag yang dipantau.
- Ekspor data — Ekspor data kapasitas ke Amazon S3 dalam format CSV atau Parquet untuk analisis lebih lanjut dan pelaporan khusus.

- API — Metrik kapasitas kueri secara terprogram menggunakan `GetCapacityManagerMetricData` dan. `GetCapacityManagerMetricDimensions`
- Pemilih tanggal — Menganalisis data kapasitas di seluruh rentang tanggal dari satu jam hingga 90 hari.
- 30+ metrik — Lacak kapasitas di seluruh Reservasi Kapasitas, On-Demand Instans, dan Instans Spot dengan lebih dari 30 metrik.

## Mengaktifkan Manajer Kapasitas EC2

Manajer Kapasitas dapat diaktifkan pada dua tingkat yang berbeda tergantung pada kebutuhan Anda. Anda dapat mengaktifkannya untuk satu orang Akun AWS untuk memantau sumber daya dalam akun itu saja, atau mengintegrasikannya dengan AWS Organisasi untuk visibilitas terpusat di semua akun anggota di organisasi Anda.

- Organization-level: Melihat dan menganalisis data kapasitas di semua akun anggota di organisasi Anda dari satu dasbor.
- Account-level: Memantau penggunaan kapasitas dalam satu Akun AWS.

Topik

- Mengaktifkan Manajer Kapasitas EC2 dengan AWS Organizations
- Mengaktifkan Manajer Kapasitas EC2 di tingkat akun
- Mendaftarkan administrator yang didelegasikan
- Menggunakan peran terkait layanan untuk Manajer Kapasitas EC2

### Mengaktifkan Manajer Kapasitas EC2 dengan AWS Organizations

Anda dapat mengaktifkan Manajer Kapasitas EC2 dengan AWS Organisasi untuk visibilitas tingkat organisasi dan pengelolaan kapasitas Anda di semua akun anggota. Integrasi ini memungkinkan Anda untuk memantau, menganalisis, dan mengelola penggunaan kapasitas dari lokasi terpusat.

Akun manajemen bertanggung jawab untuk mengaktifkan akses tingkat organisasi dan mengelola kapasitas di seluruh organisasi.

Mengaktifkan Manajer Kapasitas dengan AWS Organisasi memberikan manfaat berikut:

- Visibilitas kapasitas terpusat — Lihat penggunaan kapasitas di semua akun anggota di organisasi Anda dari satu dasbor dengan agregasi lintas akun dan lintas wilayah.
- Organization-wide optimasi — Identifikasi Reservasi Kapasitas yang tidak terpakai dan peluang pengoptimalan di semua akun di organisasi Anda.
- Administrator yang didelegasikan — Izinkan akun anggota tertentu untuk mengelola Manajer Kapasitas untuk organisasi sambil mempertahankan kontrol akses yang tepat.

Jika Anda tidak mengaktifkan integrasi dengan AWS Organisasi, Anda hanya dapat memantau sumber daya di individu Akun AWS tempat Anda mengaktifkan Manajer Kapasitas.

Prasyarat

- Anda harus memiliki pengaturan AWS Organisasi dengan akun manajemen dan satu atau lebih akun anggota. Untuk informasi selengkapnya tentang jenis akun, lihat Terminologi dan konsep di Panduan Pengguna AWS Organisasi.
- Akun manajemen harus memiliki izin untuk tindakan IAM berikut:
  - `organizations:EnableAwsServiceAccess`
  - `organizations:RegisterDelegatedAdministrator`(jika menggunakan administrasi yang didelegasikan)
  - `iam:CreateServiceLinkedRole`
- Anda harus membuat peran terkait layanan dengan kasus AWSEC2CapacityManagerServiceRolePolicy penggunaan untuk mengizinkan akses AWS Organisasi. Untuk informasi selengkapnya, lihat Membuat peran terkait layanan untuk Manajer Kapasitas.

Mengaktifkan Manajer Kapasitas dengan AWS Organizations

Menggunakan akun manajemen, aktifkan akses organisasi di Manajer Kapasitas.

Console

Untuk mengaktifkan akses organisasi di Manajer Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab Pengaturan.

4. Di bagian Ak ses tepercaya, pilih Kel ola akses tepercaya.
5. Pada prompt yang muncul, pilih Aktifkan akses tepercaya. Lalu, pilih Simpan.

AWS CLI

Untuk mengaktifkan akses organisasi di Manajer Kapasitas

1. Buat peran tertaut layanan

```
aws iam create-service-linked-role --aws-service-name
 ec2.capacitymanager.amazonaws.com
```

2. Aktif AWS kan akses Organisasi

```
aws organizations enable-aws-service-access --service-principal
 ec2.capacitymanager.amazonaws.com
```

3. Aktifkan Manajer Kapasitas dengan AWS Organisasi

```
aws ec2 enable-capacity-manager --organizations-access
```

Untuk memperbarui akses organisasi untuk Manajer Kapasitas yang ada, jalankan perintah berikut:

```
aws ec2 update-capacity-manager-organizations-access --organizations-access
```

PowerShell

Untuk mengaktifkan akses organisasi di Manajer Kapasitas

1. Buat peran terkait layanan menggunakan cmdlet. New-IAMServiceLinkedRole

```
New-IAMServiceLinkedRole -AWSServiceName "ec2.capacitymanager.amazonaws.com"
```

2. Aktif AWS kan akses Organisasi menggunakan Enable-ORGAWSServiceAccess cmdlet.

```
Enable-ORGAWSServiceAccess -ServicePrincipal "ec2.capacitymanager.amazonaws.com"
```

3. Aktifkan Manajer Kapasitas dengan AWS Organisasi menggunakan Enable-EC2CapacityManager cmdlet.

```
Enable-EC2CapacityManager -OrganizationsAccess $true
```

Untuk memperbarui akses organisasi untuk Manajer Kapasitas yang ada, jalankan Update-EC2CapacityManagerOrganizationsAccess cmdlet berikut:

```
Update-EC2CapacityManagerOrganizationsAccess -OrganizationsAccess $true
```

Memverifikasi Manajer Kapasitas diaktifkan dengan organisasi Anda

Console

Untuk memverifikasi bahwa Manajer Kapasitas diaktifkan dengan organisasi Anda

1. Di konsol Capacity Manager, pilih Peng aturan.
2. Di bagian Ak ses tepercaya, verifikasi bahwa Ak ses Organisasi ditampilkan sebagai Di aktifkan.
3. Periksa apakah ID Organisasi menampilkan ID organisasi Anda.

AWS CLI

Untuk memverifikasi bahwa Manajer Kapasitas diaktifkan dengan organisasi Anda

Jalankan perintah berikut:

```
aws ec2 get-capacity-manager-attributes
```

Output harus menampilkan:

```
{
    "CapacityManagerStatus": "enabled",
    "OrganizationsAccess": true,
    "IngestionStatus": "initial-ingestion-pending",
    "IngestionStatusMessage": "Capacity Manager is collecting historical data from
 2025-10-01T00:00:00Z. Data collection is in progress and may take several hours to
 complete."
}
```

PowerShell

Untuk memverifikasi bahwa Manajer Kapasitas diaktifkan dengan organisasi Anda

Gunakan Get-EC2CapacityManagerAttribute cmdlet.

```
Get-EC2CapacityManagerAttribute
```

Output harus menampilkan:

```
CapacityManagerStatus      : enabled
DataExportCount            : 0
EarliestDatapointTimestamp :
IngestionStatus            : initial-ingestion-in-progress
IngestionStatusMessage     : Capacity Manager is collecting historical data from
                               2026-03-17T16:00:00Z. Data collection is in progress
 and may take
                               several hours to complete.
LatestDatapointTimestamp   :
OrganizationsAccess        : True
```

Pertimbangan-pertimbangan

- Service-linked pembuatan peran: Saat Anda mengaktifkan akses organisasi melalui konsol, Manajer Kapasitas secara otomatis membuat AWSServiceRoleForEC2CapacityManager peran terkait layanan di semua akun member. Jika Anda mengaktifkan melalui AWS CLI, Anda harus menelepon `createServiceLinkedRole` secara manual.

- Agregasi data: Setelah mengaktifkan akses organisasi, Manajer Kapasitas mengisi ulang 14 hari data historis dari semua akun anggota. Proses ini biasanya memakan waktu beberapa menit untuk menyelesaikannya.
- Batasan regional: Anda hanya dapat mengaktifkan Manajer Kapasitas di satu Wilayah per organisasi, tetapi akan mengumpulkan data dari semua Wilayah komersial.
- Izin: Akun anggota tidak perlu mengambil tindakan apa pun. Manajer Kapasitas menggunakan peran terkait layanan untuk menemukan sumber daya secara otomatis di semua akun.

## Mengaktifkan Manajer Kapasitas EC2 di tingkat akun

Aktifkan Capacity Manager di tingkat akun untuk memantau dan menganalisis penggunaan kapasitas EC2 Anda dalam satu akun. AWS Setelah Anda mengaktifkannya, Capacity Manager mengumpulkan data tentang Instans, On-Demand Instans Spot, dan Reservasi Kapasitas untuk membantu Anda mengidentifikasi peluang pengoptimalan dan melacak pola penggunaan.

Aktifkan Manajer Kapasitas di tingkat akun

Console

Untuk mengaktifkan Pengelola Kapasitas untuk akun Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pada halaman Manajer Kapasitas, pilih Aktifkan di Wilayah.

AWS CLI

Untuk mengaktifkan Pengelola Kapasitas untuk akun Anda

Jalankan perintah berikut:

```
aws ec2 enable-capacity-manager
```

PowerShell

Untuk mengaktifkan Pengelola Kapasitas untuk akun Anda

Gunakan Enable-EC2CapacityManager cmdlet.

```
Enable-EC2CapacityManager
```

> **Note**
>
> - Setelah Anda mengaktifkan Manajer Kapasitas, Pengelola Kapasitas akan mengumpulkan dan mengumpulkan data historis selama 14 hari. Proses ini mungkin memakan waktu beberapa jam.
> - Saat mengumpulkan data historis Anda, `initial-ingestion-in-progress` status ditampilkan. Selama periode pengumpulan ini, Anda mungkin mengamati kesenjangan dalam data historis Anda. Ketika pengumpulan data selesai, `ingestion-complete` status ditampilkan.

## Mendaftarkan administrator yang didelegasikan

Anda dapat mendaftarkan administrator yang didelegasikan untuk Manajer Kapasitas. Ini memungkinkan akun anggota mengelola Manajer Kapasitas untuk AWS Organisasi Anda. Hanya akun manajemen yang dapat mendaftarkan atau menghapus administrator yang didelegasikan dalam organisasi Anda.

> **Note**
>
> Anda tidak dapat menonaktifkan Manajer Kapasitas untuk organisasi Anda saat ada administrator yang didelegasikan terdaftar.

Topik


- Prasyarat
- Mendaftarkan administrator yang didelegasikan
- Menghapus administrator yang didelegasikan

### Prasyarat

Akun manajemen Anda harus mengaktifkan Manajer Kapasitas dengan AWS Organisasi. Untuk informasi selengkapnya, lihat Mengaktifkan Manajer Kapasitas EC2 dengan AWS Organizations.

### Mendaftarkan administrator yang didelegasikan

Anda dapat mendaftarkan administrator yang didelegasikan menggunakan konsol Amazon EC2, the AWS CLI, or PowerShell.

#### Console

Untuk mendaftarkan administrator yang didelegasikan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab Pengaturan.
4. Di bagian Administrator yang didelegasikan, pilih Tambah.
5. Pada prompt yang muncul, masukkan ID akun anggota organisasi yang ingin Anda tambahkan sebagai administrator yang didelegasikan.
6. Pilih Tambahkan administrator yang didelegasikan.

#### AWS CLI

Untuk mendaftarkan administrator yang didelegasikan

Jalankan perintah berikut:

```
aws organizations register-delegated-administrator \
    --account-id 123456789012 \
    --service-principal ec2.capacitymanager.amazonaws.com
```

#### PowerShell

Untuk mendaftarkan administrator yang didelegasikan

Gunakan Register-ORGDelegatedAdministrator cmdlet.

```
Register-ORGDelegatedAdministrator `
    -AccountId "123456789012" `
    -ServicePrincipal "ec2.capacitymanager.amazonaws.com"
```

### Menghapus administrator yang didelegasikan

Anda dapat menghapus administrator yang didelegasikan menggunakan konsol Amazon EC2, the AWS CLI, or PowerShell.

#### Console

Untuk menghapus administrator yang didelegasikan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab Pengaturan.
4. Di bagian Administrator yang didelegasikan, pilih Kelola.
5. Pada prompt yang muncul, pilih Hapus administrator yang didelegasikan.

#### AWS CLI

Untuk menghapus administrator yang didelegasikan

Jalankan perintah berikut:

```
aws organizations deregister-delegated-administrator \
    --account-id 123456789012 \
    --service-principal ec2.capacitymanager.amazonaws.com
```

#### PowerShell

Untuk menghapus administrator yang didelegasikan

Gunakan Unregister-ORGDelegatedAdministrator cmdlet.

```
Unregister-ORGDelegatedAdministrator `
    -AccountId "123456789012" `
```

```
    -ServicePrincipal "ec2.capacitymanager.amazonaws.com"
```

## Menggunakan peran terkait layanan untuk Manajer Kapasitas EC2

Manajer Kapasitas EC2 menggunakan peran AWS Identity and Access Management terkait layanan (IAM). Peran terkait layanan adalah jenis peran IAM unik yang ditautkan langsung ke Manajer Kapasitas. Service-linked peran telah ditentukan sebelumnya oleh Manajer Kapasitas dan menyertakan semua izin yang diperlukan layanan untuk memanggil AWS layanan lain atas nama Anda.

Peran terkait layanan membuat pengaturan Capacity Manager lebih mudah karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. Manajer Kapasitas mendefinisikan izin peran terkait layanan, dan kecuali ditentukan lain, hanya Manajer Kapasitas yang dapat mengambil perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, dan kebijakan izin tersebut tidak dapat dilampirkan ke entitas IAM lainnya.

Anda dapat menghapus peran tertaut layanan hanya setelah menghapus sumber daya terkait terlebih dahulu. Ini melindungi sumber daya Manajer Kapasitas Anda karena Anda tidak dapat menghapus izin untuk mengakses sumber daya secara tidak sengaja.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, lihat AWS layanan yang bekerja dengan IAM dan cari layanan yang memiliki Ya di kolom peran. Service-linked Pilih Ya dengan sebuah tautan untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Service-linked izin peran untuk Manajer Kapasitas

Manajer Kapasitas menggunakan peran terkait layanan yang diberi nama AWSServiceRoleForEC2CapacityManager untuk memungkinkan Anda mengelola sumber daya kapasitas dan mengintegrasikan dengan AWS Organisasi atas nama Anda.

Per AWSServiceRoleForEC2CapacityManager an terkait layanan mempercayai layanan berikut untuk mengambil peran:

- `ec2.capacitymanager.amazonaws.com`

Kebijakan izin peran yang dinam AWSEC2CapacityManagerServiceRolePolicy ai memungkinkan Manajer Kapasitas menyelesaikan tindakan berikut:

- `organizations:DescribeOrganization`

- `organizations:ListAccounts`
- `organizations:ListChildren`
- `organizations:ListAWSServiceAccessForOrganization`
- `organizations:ListDelegatedAdministrators`

Anda harus mengonfigurasi izin untuk mengizinkan pengguna, grup, atau peran Anda membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat izin Service-linked peran di Panduan Pengguna IAM.

Membuat peran terkait layanan untuk Manajer Kapasitas

Anda dapat menggunakan konsol IAM untuk membuat peran terkait layanan dengan kasus penggunaan. AWSEC2CapacityManagerServiceRolePolicy Di AWS API AWS CLI atau, buat peran terkait layanan dengan nama `ec2.capacitymanager.amazonaws.com` layanan. Untuk informasi selengkapnya, lihat Membuat peran tertaut layanan dalam Panduan Pengguna IAM. Jika Anda menghapus peran tertaut layanan ini, Anda dapat mengulang proses yang sama untuk membuat peran tersebut lagi.

Mengedit peran terkait layanan untuk Manajer Kapasitas

Manajer Kapasitas tidak mengizinkan Anda mengedit AWSServiceRoleForEC2CapacityManager peran terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin merujuk peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran terkait layanan dalam Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk Manajer Kapasitas

Jika Anda tidak perlu lagi menggunakan fitur atau layanan yang memerlukan peran terkait layanan, kami merekomendasikan Anda menghapus peran tersebut. Dengan begitu, Anda tidak memiliki entitas yang tidak digunakan yang tidak dipantau atau dipelihara secara aktif. Tetapi, Anda harus membersihkan sumber daya peran terkait layanan sebelum menghapusnya secara manual.

> **Note**
>
> Jika layanan Manajer Kapasitas menggunakan peran saat Anda mencoba menghapus sumber daya, penghapusan mungkin gagal. Jika hal itu terjadi, tunggu beberapa menit dan coba mengoperasikannya lagi.

Untuk menghapus sumber daya Capacity Manager yang digunakan oleh AWSServiceRoleForEC2CapacityManager

1. Semua administrator yang didelegasikan harus menonaktifkan Manajer Kapasitas mereka sebelum menghapus akses organisasi.
2. Anda harus menghapus ekspor data aktif sebelum menonaktifkan pengelola kapasitas.

Untuk menghapus peran tertaut layanan secara manual menggunakan IAM

Gunakan konsol IAM, AWS CLI, atau AWS API untuk menghapus peran terkait AWSServiceRoleForEC2CapacityManager layanan. Untuk informasi selengkapnya, lihat Menghapus peran terkait layanan dalam Panduan Pengguna IAM.

Wilayah yang Didukung untuk peran terkait layanan Manajer Kapasitas

Capacity Manager mendukung penggunaan peran terkait layanan di semua Wilayah tempat layanan tersedia. Untuk informasi selengkapnya, lihat AWS Wilayah dan titik akhir.

Manajer Kapasitas tidak mendukung penggunaan peran terkait layanan di setiap Wilayah tempat layanan tersedia. Anda dapat menggunakan AWSServiceRoleForEC2CapacityManager peran di Wilayah berikut.

| Nama wilayah | Identitas wilayah | Dukungan dalam Capacity Manager |
|---|---|---|
| US East (Northern Virginia) | us-east-1 | Ya |
| US East (Ohio) | us-east-2 | Ya |
| US West (N. California) | us-west-1 | Ya |
| US West (Oregon) | us-west-2 | Ya |
| Africa (Cape Town) | af-south-1 | Tidak |
| Asia Pasifik (Hong Kong) | ap-east-1 | Tidak |
| Asia Pasifik (Jakarta) | ap-southeast-3 | Tidak |
| Asia Pasifik (Mumbai) | ap-south-1 | Ya |

| Nama wilayah | Identitas wilayah | Dukungan dalam Capacity Manager |
|---|---|---|
| Asia Pacific (Osaka) | ap-northeast-3 | Ya |
| Asia Pacific (Seoul) | ap-northeast-2 | Ya |
| Asia Pacific (Singapore) | ap-southeast-1 | Ya |
| Asia Pacific (Sydney) | ap-southeast-2 | Ya |
| Asia Pacific (Tokyo) | ap-northeast-1 | Ya |
| Canada (Central) | ca-sentral-1 | Ya |
| Eropa (Frankfurt) | eu-central-1 | Ya |
| Eropa (Irlandia) | eu-west-1 | Ya |
| Eropa (London) | eu-west-2 | Ya |
| Europe (Milan) | eu-south-1 | Tidak |
| Eropa (Paris) | eu-west-3 | Ya |
| Eropa (Stockholm) | eu-north-1 | Ya |
| Timur Tengah (Bahrain) | me-south-1 | Tidak |
| Timur Tengah (UAE) | me-central-1 | Tidak |
| Amerika Selatan (Sao Paulo) | sa-east-1 | Ya |
| AWS GovCloud (US-East) | us-gov-east-1 | Tidak |
| AWS GovCloud (US-West) | us-gov-west-1 | Tidak |

# Mengatur data Anda di Capacity Manager

Capacity Manager menggunakan kombinasi metrik, titik data, dimensi, rentang tanggal, dan periode untuk mengatur data kapasitas Anda. Ini dapat membantu Anda menganalisis pola penggunaan dan membuat keputusan berdasarkan informasi tentang sumber daya Anda.

Metrik dan titik data

Metrik adalah kumpulan titik data yang diatur waktu. Misalnya, jika Anda ingin memantau penggunaan Spot di vCPU, Anda akan menggunakan `SpotTotalUsageHrsVcpu` metrik.

Setiap jam, metrik menghasilkan titik data stempel waktu dengan penggunaan Spot dalam jam vCPU. Misalnya, jika Anda menggunakan 100 vCPU selama jam 10:00 AM, Manajer Kapasitas membuat titik data dengan stempel waktu 10:00 AM dan nilai 100.

Untuk daftar lengkap metrik yang dianalisis oleh Manajer Kapasitas, lihatMetrik Manajer Kapasitas EC2.

Dimensi

Dimensi adalah pasangan nama-nilai yang membantu Anda mengkategorikan dan mengidentifikasi berbagai aspek metrik. Misalnya, nama satu dimensi di Capacity Manager adalah accountID, di mana nilainya adalah ID akun aktual. Capacity Manager menyediakan dimensi untuk mengelompokkan dan mengelompokkan data Anda seperti Keluarga Instans, Reservasi ARN, Jenis Reservasi, dan Tenancy.

Selain dimensi bawaan, Anda dapat mengaktifkan kunci tag dari sumber daya Amazon EC2 untuk digunakan sebagai dimensi khusus. Misalnya, jika instance Anda diberi tag dengan `environment` atau`team`, Anda dapat mengaktifkan kunci tag tersebut lalu mengelompokkan dan memfilter metrik kapasitas berdasarkan nilainya.

Capacity Manager juga menyediakan Manager-provided tag Kapasitas — seperti EC2 Auto Scaling Group (`aws:autoscaling:groupName`), EKS cluster name (`aws:eks:cluster-name`), EKS Kubernetes node pool (`eks:kubernetes-node-pool-name`), dan Karpenter node pool (`karpenter.sh/nodepool`) — yang tersedia secara default tanpa menghitung batas kunci tag Anda.

Untuk daftar lengkap dimensi, lihatMetrik Manajer Kapasitas EC2. Untuk informasi tentang mengaktifkan dan mengelola dimensi tag, lihatMengelola kunci tag yang dipantau.

Rentang tanggal dan periode

Rentang tanggal menentukan berapa banyak waktu yang ingin Anda analisis, dari satu jam hingga 90 hari. Periode menentukan bagaimana Capacity Manager mengumpulkan data Anda sepanjang waktu dan berapa banyak titik data yang akan dikembalikan. Misalnya, jika rentang tanggal Anda adalah satu minggu dan menstruasi Anda adalah 1 hari, Manajer Kapasitas mengembalikan 7 titik data. Setiap titik data mewakili satu hari data agregat. Periode harus selang waktu satu jam dan dibagi secara merata ke dalam rentang tanggal.

Topik


- Metrik Manajer Kapasitas EC2
- Pengelompokan dan memfilter data
- Mengelola kunci tag yang dipantau


## Metrik Manajer Kapasitas EC2

Capacity Manager menawarkan pilihan metrik yang komprehensif untuk melacak kapasitas Anda di berbagai jenis sumber daya. Metrik dapat diukur menggunakan unit yang berbeda tergantung pada kebutuhan analisis Anda.

Nama metrik di Capacity Manager menggunakan empat awalan berbeda untuk mengkategorikan jenis kapasitas yang diukur:

- `Reservation`— Reservasi Kapasitas itu sendiri, termasuk total kapasitas yang dipesan, tingkat pemanfaatan, kapasitas yang tidak digunakan, dan jumlah reservasi.
- `Reserved`— Penggunaan On-Demand instans yang dicakup oleh Reservasi Kapasitas Anda.
- `Unreserved`— Penggunaan On-Demand instans yang berjalan di luar Reservasi Kapasitas apa pun.
- `Spot`— Khusus untuk penggunaan Instans Spot, termasuk runtime dan perkiraan biaya. Metrik ini terpisah dari kapasitas berbasis reservasi.

Tabel berikut juga menyediakan  Dimensi yang tersedia  untuk setiap metrik. Kategori dimensi dipecah sebagai berikut:

- Dimensi kapasitas umum  — ID Akun, Nama Akun, Wilayah, Keluarga Instans, Zona Ketersediaan, Jenis Instans, Platform, dan Sewa

- Dimensi Reservasi Kapasitas — ID Reservasi (ID Reservasi Kapasitas), ARN Reservasi, Pemilik Keuangan yang Tidak Digunakan, Jenis Reservasi (ODCR/Capacity Blok), Buat stempel waktu, Stempel waktu Mulai, Stempel waktu Akhir, Status, dan kriteria kecocokan Instans.
- Dimensi penggunaan yang di pesan — Reservasi ID (CRID), Reservasi ARN, Jenis reservasi
- Dimensi tag — kunci Customer-managed tag dan Manager-provided tag Kapasitas. Anda harus mengaktifkan kunci tag sebelum Anda dapat menggunakannya sebagai dimensi. Untuk informasi selengkapnya, lihat Mengelola kunci tag yang dipantau.

| Metrik | Deskripsi | Dimensi tersedia | Unit tersedia |
|---|---|---|---|
| `ReservationAvgCommittedSize` | Jumlah rata-rata total kapasitas dalam keadaan aktif atau terjadwal dengan komitmen. Ukuran dijumlahkan lintas dimensi dan dirata-ratakan dari waktu ke waktu. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationAvgFutureSize` | Jumlah rata-rata Reservasi Kapasitas yang dijadwalkan untuk dimulai di masa mendatang tetapi belum aktif selama periode yang dipilih. Ukuran dijumlahkan lintas dimensi dan dirata-ratakan dari waktu ke waktu. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationAvgUtilization` | Persentase rata-rata kapasitas cadangan yang digunakan selama periode yang dipilih. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationMaxCommittedSize` | Kapasitas total maksimum dalam keadaan aktif atau terjadwal dengan komitmen. Ukuran dijumlahkan di seluruh | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |

| Metrik | Deskripsi | Dimensi tersedia | Unit tersedia |
| --- | --- | --- | --- |
| | dimensi dan nilai maksimum diambil untuk periode tersebut. | | |
| ReservationMaxFutureSize | Jumlah maksimum Reservasi Kapasitas yang dijadwalkan untuk dimulai di masa mendatang tetapi belum aktif selama periode yang dipilih. Ukuran dijumlahkan di seluruh dimensi dan nilai maksimum diambil untuk periode tersebut. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| ReservationMaxSize | Ukuran maksimum yang dicapai Reservasi Kapasitas Anda setiap saat selama periode yang dipilih. Diperlukan ID Reservasi. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| ReservationMaxUnusedSize | Jumlah maksimum kapasitas yang tidak terpakai dalam Reservasi Kapasitas Anda setiap saat selama periode yang dipilih. Diperlukan ID Reservasi. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| ReservationMaxUtilization | Persentase pemanfaatan maksimum yang dicapai Reservasi Kapasitas Anda setiap saat selama periode yang dipilih. Diperlukan ID Reservasi. | Kapasitas umum dan Dimensi Reservasi Kapasitas | |
| ReservationMinCommittedSize | Jumlah total kapasitas minimum dalam keadaan aktif atau terjadwal dengan komitmen. Ukuran dijumlahkan di seluruh dimensi dan nilai minimum diambil untuk periode tersebut. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |

| Metrik | Deskripsi | Dimensi tersedia | Unit tersedia |
| --- | --- | --- | --- |
| `ReservationMinFutureSize` | Jumlah minimum Reservasi Kapasitas yang dijadwalkan untuk dimulai di masa mendatang tetapi belum aktif pada titik mana pun selama periode yang dipilih. Ukuran dijumlahkan di seluruh dimensi dan nilai minimum diambil untuk periode tersebut. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationMinSize` | Ukuran minimum yang dicapai Reservasi Kapasitas Anda setiap saat selama periode yang dipilih. Diperlukan ID Reservasi. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationMinUnusedSize` | Jumlah minimum kapasitas yang tidak terpakai dalam Reservasi Kapasitas Anda setiap saat selama periode yang dipilih. Diperlukan ID Reservasi. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationMinUtilization` | Persentase pemanfaatan minimum yang dicapai Reservasi Kapasitas Anda setiap saat selama periode yang dipilih. Diperlukan ID Reservasi. | Kapasitas umum dan Dimensi Reservasi Kapasitas | |
| `ReservationTotalCapacityHrs` | Jumlah total kapasitas yang telah Anda pesan melalui Reservasi Kapasitas selama periode yang dipilih. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationTotalCount` | Jumlah total Reservasi Kapasitas berbeda yang Anda miliki selama periode yang dipilih. | Kapasitas umum dan Dimensi Reservasi Kapasitas | |

| Metrik | Deskripsi | Dimensi tersedia | Unit tersedia |
| --- | --- | --- | --- |
| `ReservationTotalEstimatedCost` | Perkiraan biaya total jam kapasitas yang dipesan selama periode yang dipilih. | Kapasitas umum dan Dimensi Reservasi Kapasitas | |
| `ReservationUnusedTotalCapacityHrs` | Jumlah total kapasitas cadangan yang Anda miliki tetapi tidak digunakan selama periode yang dipilih. | Kapasitas umum dan Dimensi Reservasi Kapasitas | vCPU, Instans |
| `ReservationUnusedTotalEstimatedCost` | Perkiraan biaya kapasitas cadangan yang Anda miliki tetapi tidak digunakan selama periode yang dipilih (dihitung menggunakan On-Demand tarif). | Kapasitas umum dan Dimensi Reservasi Kapasitas | |
| `ReservedTotalEstimatedCost` | Perkiraan biaya penggunaan On-Demand Instans yang ditanggung oleh Reservasi Kapasitas selama periode yang dipilih. Ini tidak termasuk penggunaan Spot. | Kapasitas umum dan dimensi penggunaan Cadangan | |
| `ReservedTotalUsageHrs` | Total jam penggunaan On-Demand Instans yang dicakup oleh Reservasi Kapasitas selama periode yang dipilih. Ini tidak termasuk penggunaan Spot. | Kapasitas umum dan dimensi penggunaan Cadangan | vCPU, Instans |
| `SpotAvgRunTimeBeforeInterruption` | Runtime rata-rata dalam jam untuk instans yang terputus dalam periode yang dipilih. | Dimensi Wilayah, AZ, dan ID Akun saja | Instans |

| Metrik | Deskripsi | Dimensi tersedia | Unit tersedia |
| --- | --- | --- | --- |
| SpotInterruptionRate | Persentase menjalankan Instans Spot yang terganggu dalam periode yang dipilih. | Dimensi Wilayah, AZ, dan ID Akun saja | vCPU, Instans |
| SpotMaxRunTimeBeforeInterruption | Waktu proses maksimum dalam jam untuk instans yang terputus dalam periode yang dipilih. | Dimensi Wilayah, AZ, dan ID Akun saja | Instans |
| SpotMinRunTimeBeforeInterruption | Waktu proses minimum dalam jam untuk instans yang terputus dalam periode yang dipilih. | Dimensi Wilayah, AZ, dan ID Akun saja | Instans |
| SpotTotalCount | Jumlah Instans Spot atau vCPU yang berjalan selama periode yang dipilih. | Dimensi Wilayah, AZ, dan ID Akun saja | vCPU, Instans |
| SpotTotalEstimatedCost | Perkiraan biaya penggunaan Instans Spot selama periode yang dipilih (dihitung menggunakan tarif Spot yang dipublikasikan). | Dimensi kapasitas umum | |
| SpotTotalInterruptions | Jumlah Instans Spot atau vCPU yang terputus selama periode yang dipilih. | Dimensi Wilayah, AZ, dan ID Akun saja | vCPU, Instans |
| SpotTotalUsageHrs | Total jam penggunaan Instans Spot selama periode yang dipilih. | Dimensi kapasitas umum | vCPU, Instans |
| UnreservedTotalEstimatedCost | Perkiraan biaya penggunaan On-Demand Instans yang tidak tercakup oleh Reservasi Kapasitas selama periode yang dipilih. Ini tidak termasuk penggunaan Spot. | Dimensi kapasitas umum | |

| Metrik | Deskripsi | Dimensi tersedia | Unit tersedia |
| --- | --- | --- | --- |
| `UnreservedTotalUsageHrs` | Total jam penggunaan On-Demand Instans yang tidak tercakup oleh Reservasi Kapasitas selama periode yang dipilih. Ini tidak termasuk penggunaan Spot. | Dimensi kapasitas umum | vCPU, Instans |

> **Note**
>
> Jika Anda menyertakan instance dalam unit Anda, sebaiknya sertakan jenis instans dalam dimensi Anda.

Dimensi tag

Selain dimensi bawaan di bagian sebelumnya, Capacity Manager mendukung dimensi tag. Dimensi tag memungkinkan Anda mengelompokkan dan memfilter metrik menggunakan kunci tag dari sumber daya Amazon EC2 Anda.

Customer-managed dimensi tag

Anda dapat mengaktifkan hingga lima kunci tag untuk digunakan sebagai dimensi. Setelah aktivasi dan dalam `activated` status, dimensi tag tersedia untuk metrik yang mendukung dimensi kapasitas Umum. Untuk daftar lengkap kategori dimensi, lihatMetrik Manajer Kapasitas EC2.

| Dimensi | Deskripsi | Contoh nilai |
| --- | --- | --- |
| Nama kunci tag | Kunci tag yang dikelola pelanggan dari sumber daya Amazon EC2 Anda. | `environment` , `team`, `cost-center` |

Dim Manager-provided ensi tag kapasitas

Manajer Kapasitas menyediakan Manager-provided tag Kapasitas berikut secara default. Manager-provided Tag kapasitas selalu tersedia dan tidak dihitung dalam batas kunci tag Anda.

| Dimensi | Deskripsi |
| --- | --- |
| `tag:aws:autoscaling:groupName` | Nama grup Penskalaan Otomatis EC2 yang terkait dengan instance. |
| `tag:aws:eks:cluster-name` | Nama cluster EKS yang terkait dengan instance. |
| `tag:eks:kubernetes-node-pool-name` | Kumpulan node EKS Kubernetes terkait dengan instance. |
| `tag:karpenter.sh/nodepool` | Kumpulan node Karpenter yang terkait dengan instance. |

> **Note**
>
> Saat Anda mengelompokkan berdasarkan dimensi tag, sumber daya yang tidak memiliki nilai untuk tag tersebut disertakan dalam bucket terpisah dengan nilai string kosong. Ini memastikan bahwa total memperhitungkan semua sumber daya.

## Pengelompokan dan memfilter data

Manajer Kapasitas menggabungkan metrik Anda berdasarkan dimensi dan periode tanggal yang Anda pilih. Jika tidak ada dimensi yang dipilih, Manajer Kapasitas akan mengumpulkan data dan mengembalikan satu titik data per periode dalam rentang tanggal. Anda dapat mengelompokkan data menjadi agregasi yang lebih kecil dengan menambahkan dimensi, termasuk dimensi tag dari sumber daya Amazon EC2 Anda.

- Peng  elompokan — Pisahkan data kapasitas Anda berdasarkan dimensi seperti Wilayah, Keluarga Instans, ID Akun, atau kunci tag. Anda dapat mengelompokkan metrik berdasarkan beberapa dimensi untuk memecah data Anda lebih lanjut. Misalnya, jika Anda mengelompokkan berdasarkan Wilayah dan Zona Ketersediaan, Anda mendapatkan titik data untuk setiap kombinasi Wilayah dan AZ tempat Anda menggunakan.
- Penyaringan  — Tampilkan hanya subset tertentu dari dimensi yang Anda pilih. Misalnya, jika Anda mengelompokkan berdasarkan keluarga instance, Anda akan mendapatkan titik data untuk semua keluarga tempat Anda menggunakan. Namun, jika Anda juga memfilter berdasarkan p5, Anda hanya melihat titik data untuk keluarga instans p5. Anda juga dapat memfilter berdasarkan nilai

tag — misalnya, memfilter berdasarkan `environment=prod` untuk hanya melihat sumber daya produksi. Untuk memfilter sumber daya yang tidak memiliki nilai untuk tag yang dipilih, filter dengan nilai kosong.

- Unit metrik — Lihat hasil berdasarkan unit yang berbeda seperti vCPU, instans, atau perkiraan biaya. Misalnya, setelah mengelompokkan berdasarkan Wilayah dan memfilter menurut keluarga instans tertentu, Anda dapat beralih antara total vCPU yang digunakan, jumlah instans yang berjalan, atau perkiraan biaya.

Cara mengelompokkan dan memfilter data di konsol

Untuk mengelompokkan dan memfilter data di Manajer Kapasitas

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab untuk jenis sumber daya yang ingin Anda analisis: Penggunaan, Reservasi, atau Spot.
4. Di bagian Filter Tanggal, pilih Rentang tanggal, Zona waktu, dan Interval.
5. Di bagian Filter Dimensi, pilih dimensi dari dropdown Pilih dimensi. Opsi dimensi berbeda untuk setiap jenis sumber daya. Untuk informasi selengkapnya, lihat Metrik Manajer Kapasitas EC2.

   Untuk mengelompokkan berdasarkan dimensi tag, pilih kunci tag dari dropdown (misalnya,`environment`). Hanya kunci tag dalam `activated` status yang tersedia. Untuk informasi selengkapnya tentang mengaktifkan kunci tag, lihatMengelola kunci tag yang dipantau.

   Anda dapat menambahkan beberapa dimensi, termasuk beberapa dimensi tag, untuk membuat pengelompokan yang lebih terperinci.
6. Untuk memfilter berdasarkan dimensi yang Anda pilih, pilih opsi filter dari tarik- turun Filter berdasarkan dimensi.

   Untuk dimensi tag, filter menunjukkan nilai tag yang ada dalam data Anda. Untuk memfilter sumber daya tanpa nilai untuk tag yang dipilih, pilih opsi nilai kosong.
7. Di bagian Agregasi, pilih unit untuk melihat hasil berdasarkan vCPU, Instans, atau Perkiraan biaya.

> **Note**
>
> Saat Anda mengelompokkan berdasarkan dimensi tag, sumber daya yang tidak memiliki nilai untuk tag tersebut disertakan dalam grup terpisah dengan nilai kosong. Ini memastikan bahwa total memperhitungkan semua sumber daya, tidak hanya yang ditandai.

## Mengelola kunci tag yang dipantau

Amazon EC2 Capacity Manager memungkinkan Anda memilih kunci tag dari sumber daya Amazon EC2 Anda (misalnya, `environment` atau`team`) untuk digunakan sebagai dimensi saat menganalisis data kapasitas Anda. Setelah kunci tag diaktifkan, Anda dapat mengelompokkan dan memfilter metrik berdasarkan nilai tag tersebut — seperti halnya dengan Wilayah, Jenis Instans, atau Zona Ketersediaan.

Setiap akun dapat memantau hingga lima kunci tag. Manager-provided Tag kapasitas tidak dihitung dalam batas ini.

Topik


- Manager-provided Tag kapasitas
- Lifecycle tag
- Aktifkan dan menonaktifkan kunci tag yang dipantau
- Lihat kunci tag yang dipantau
- Metrik kueri menggunakan dimensi tag
- Tag dalam ekspor data
- Organisasi dan administrator yang didelegasikan
- Pertimbangan-pertimbangan


### Manager-provided Tag kapasitas

Capacity Manager menyertakan satu set tag default untuk setiap akun. Ini tidak dihitung dalam batas kunci tag Anda. Manager-provided Tag kapasitas mewakili dimensi pengelompokan yang umum digunakan dan meliputi:

- `aws:autoscaling:groupName`- Grup Penskalaan Otomatis EC2
- `aws:eks:cluster-name`— Nama cluster EKS

- `eks:kubernetes-node-pool-name`— Kumpulan simpul EKS Kubernetes
- `karpenter.sh/nodepool`— Kumpulan simpul Karpenter

Manager-provided Tag kapasitas muncul `GetCapacityManagerMonitoredTagKeys` dengan diset `CapacityManagerProvided` el ke `true` dan tidak dapat diaktifkan atau dinonaktifkan oleh pelanggan. Saat Manajer Kapasitas pertama kali diaktifkan, Manager-provided tag Kapasitas mulai dalam `activating` status dan beralih ke `activated` setelah Manajer Kapasitas menerima titik data pertama yang menyertakan Manager-provided tag Kapasitas, biasanya dalam satu hingga dua jam.

Lifecycle tag

Kunci tag yang dipantau berkembang melalui status berikut:

| Status | Deskripsi |
| --- | --- |
| `activating` | Kunci tag terdaftar. Manajer Kapasitas sedang mempersiapkan untuk mengumpulkan data untuk tag ini. Anda tidak dapat menanyakan metrik menggunakan tag dalam status ini. |
| `activated` | Data tag sedang dicerna dan dapat dikueri melalui API metrik dan dalam ekspor data. |
| `suspended` | Kunci tag telah melampaui ambang 100.000 nilai tag unik. Tag masih diperhitungkan dalam batas Anda, tetapi Manajer Kapasitas tidak lagi menyerap data untuk itu. Jika penggunaan nilai tag Anda menurun di bawah ambang batas untuk jangka waktu yang berkelanjutan, tag diaktifkan kembali secara otomatis. |
| `deactivating` | Kunci tag sedang dihapus. Ini tidak akan muncul lagi `GetCapacityManagerMonitoredTagKeys` setelah penonaktifan selesai. |

Saat tag dalam `suspended` status, `GetCapacityManagerMonitoredTagKeys` mengembalikan pesan status berikut: “Tag ditangguhkan karena terlalu banyak nilai tag. Kurangi penggunaan tag atau nonaktifkan.”

> **Note**
>
> Jika Anda menonaktifkan kunci tag dan kemudian mengaktifkan kembali kunci yang sama, hanya data yang dicerna setelah pengaktifan ulang yang dapat dikueri. Data historis dari aktivasi sebelumnya tidak dapat diakses. `EarliestDatapointTimestamp`Reset dengan setiap aktivasi baru.

### Aktifkan dan menonaktifkan kunci tag yang dipantau

Anda dapat mengaktifkan kunci tag untuk mulai memantaunya sebagai dimensi, atau menonaktifkan kunci tag yang tidak lagi Anda butuhkan. Aktivasi tidak sinkron — tag memasuki `activating` status dan beralih ke `activated` setelah Capacity Manager mulai menerima data untuk tag itu. Penonaktifan menghapus kunci tag dari set yang dipantau.

> **Note**
>
> Saat mengaktifkan kunci tag, masukkan hanya nama kunci (misalnya,`environment`). Capacity Manager secara otomatis membuatnya tersedia sebagai dimensi untuk mengelompokkan dan memfilter metrik Anda.

#### Console

Untuk mengaktifkan atau menonaktifkan kunci tag yang dipantau

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab Pengaturan.
4. Di bagian K unci tag yang dipantau, pilih Kelola kunci tag.
5. Untuk mengaktifkan kunci tag, masukkan nama kunci tag dan pilih Tam bah. Untuk menonaktifkan kunci tag, pilih kunci tag dan pilih H apus.
6. Pilih Simpan perubahan.

#### AWS CLI

Untuk mengaktifkan kunci tag

Gunakan perintah berikut untuk mengaktifkan satu atau beberapa kunci tag:

```
aws ec2 update-capacity-manager-monitored-tag-keys \
    --activate-tag-keys "environment" "teamId"
```

Output menunjukkan kunci tag dan status awalnya:

```
{
    "CapacityManagerTagKeys": [
        {
            "TagKey": "environment",
            "Status": "activating"
        },
        {
            "TagKey": "teamId",
            "Status": "activating"
        }
    ]
}
```

Untuk menonaktifkan kunci tag

Gunakan perintah berikut untuk menonaktifkan satu atau beberapa kunci tag:

```
aws ec2 update-capacity-manager-monitored-tag-keys \
    --deactivate-tag-keys "project"
```

Output menunjukkan status yang diperbarui:

```
{
    "CapacityManagerTagKeys": [
        {
            "TagKey": "project",
            "Status": "deactivating"
        }
    ]
}
```

Anda dapat mengaktifkan dan menonaktifkan kunci tag dalam permintaan yang sama:

```
aws ec2 update-capacity-manager-monitored-tag-keys \
    --activate-tag-keys "environment" "teamId" \
    --deactivate-tag-keys "project"
```

Lihat kunci tag yang dipantau

Anda dapat melihat semua kunci tag yang dipantau untuk akun Anda, termasuk statusnya saat ini dan stempel waktu paling awal dari mana data tersedia.

Console

Untuk melihat kunci tag yang dipantau

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab Pengaturan.
4. Di bagian Kunci tag yang dipantau, lihat kunci tag Anda, statusnya, dan apakah itu Manager-provided tag Kapasitas.

AWS CLI

Untuk melihat kunci tag yang dipantau

Jalankan perintah berikut:

```
aws ec2 get-capacity-manager-monitored-tag-keys
```

Output mencantumkan semua kunci tag, termasuk tag Kapasitas dan Manager-provided tag yang dikelola pelanggan:

```
{
    "CapacityManagerTagKeys": [
        {
```

```
            "TagKey": "aws:autoscaling:groupName",
            "Status": "activated",
            "CapacityManagerProvided": true,
            "earliestDatapointTimestamp": "2026-04-08T00:00:00"
        },
        {
            "TagKey": "environment",
            "Status": "activated",
            "CapacityManagerProvided": false,
            "earliestDatapointTimestamp": "2025-08-11T22:00:00"
        }
    ]
}
```

Bid `CapacityManagerProvided` ang menunjukkan apakah tag adalah tag Kapasitas (`true`) atau Manager-provided tag yang dikelola pelanggan (`false`). Manager-provided Tag kapasitas tidak dihitung dalam batas kunci tag Anda. Ini `EarliestDatapointTimestamp` menunjukkan titik waktu paling awal dari mana data tersedia untuk kunci tag tersebut.

Metrik kueri menggunakan dimensi tag

Setelah kunci tag mencapai `activated` status, Anda dapat menggunakannya sebagai dimensi di `GetCapacityManagerMetricDimensions` dan`GetCapacityManagerMetricData`.

Untuk menanyakan metrik yang dikelompokkan berdasarkan dimensi tag

Gunakan perintah berikut ini.

```
aws ec2 get-capacity-manager-metric-dimensions \
    --group-by tag:environment account-id \
    --filter-by
 'DimensionCondition={Dimension=tag:environment,Comparison=equals,Values=[prod]}'
```

Saat Anda mengelompokkan berdasarkan dimensi tag, hasilnya mencakup semua sumber daya di akun Anda — tidak hanya sumber daya yang memiliki tag. Sumber daya yang tidak memiliki nilai untuk tag dikelompokkan ke dalam bucket terpisah dengan nilai string kosong. Misalnya, jika akun Anda menggunakan 800 jam vCPU dalam periode tertentu dan hanya beberapa sumber daya tersebut yang memiliki `environment` tag, pengelompokan berdasarkan kunci `environment` tag mungkin muncul:

- `prod`— 300 jam vCPU
- `staging`— 200 jam vCPU
- `""`(string kosong) - 300 jam vCPU dari sumber daya tanpa tag `environment`

Ini memastikan bahwa total di semua bucket memperhitungkan penggunaan penuh Anda. Anda dapat secara eksplisit memfilter sumber daya yang tidak ditandai dengan meneruskan string kosong sebagai nilai filter:

```
--filter-by
 'DimensionCondition={Dimension=tag:environment,Comparison=equals,Values=[""]}'
```

> **Note**
>
> Jika Anda menanyakan dengan kunci tag yang masih dalam `activating` status, kueri ditolak dengan kesalahan 400. Tunggu status tag berubah menjadi `activated` sebelum melakukan query. Anda dapat memeriksa status menggunakan`GetCapacityManagerMonitoredTagKeys`.

> **Note**
>
> Kueri dengan waktu mulai sebelum dimensi tag yang disediakan ditolak. `EarliestDatapointTimestamp` Gunakan `GetCapacityManagerMonitoredTagKeys` untuk memeriksa kapan data tersedia untuk setiap tag.

Tag dalam ekspor data

Saat Anda mengaktifkan pemantauan tag, ekspor data Anda menyertakan kunci tag yang diaktifkan dan Manager-provided tag Kapasitas sebagai kolom tambahan. Kolom tag muncul setelah semua kolom standar dengan header seperti `tag:environment` dan`tag:team`. Kolom tag diurutkan berdasarkan abjad.

Ekspor hanya menyertakan tag dalam `activated` status. Manajer Kapasitas mengecualikan tag di `activating`,`deactivating`, atau `suspended` status.

> **Note**
>
> Jika Anda mengaktifkan kunci tag baru, ekspor data yang ada tidak secara otomatis menyertakan tag baru. Anda harus membuat ekspor data baru untuk menyertakan kunci tag yang baru diaktifkan sebagai kolom.

Organisasi dan administrator yang didelegasikan

Jika akun Anda adalah bagian dari AWS Organisasi dengan Manajer Kapasitas tingkat organisasi diaktifkan, setiap akun (administrator organisasi dan administrator yang didelegasikan) dapat secara independen mengaktifkan, menonaktifkan, dan menanyakan kunci tag. Setiap akun mempertahankan status tag sendiri`EarliestDatapointTimestamp`, dan batas kunci tag.

Akun hanya dapat menanyakan data metrik untuk kunci tag yang telah diaktifkan oleh akun itu sendiri. Jika administrator organisasi dan administrator yang didelegasikan mengaktifkan kunci tag yang sama (misalnya,`environment`), setiap akun melacak status aktivasi dan ketersediaan datanya sendiri secara independen.

Ketika administrator yang didelegasikan menonaktifkan kunci tag, administrator yang didelegasikan tidak dapat lagi menanyakan data untuk tag tersebut, bahkan jika administrator organisasi masih mengaktifkan kunci tag yang sama.

Pertimbangan-pertimbangan

- Pembaruan nilai tag:  Nilai tag untuk sumber daya baru dan tag yang baru diterapkan ke sumber daya yang ada tersedia dalam beberapa jam. Jika Anda mengubah nilai tag yang ada pada sumber daya, nilai yang diperbarui dapat memakan waktu hingga 24 jam untuk mencerminkan di Manajer Kapasitas.
- Waktu aktivasi:  Setelah Anda mengaktifkan kunci tag, diperlukan waktu hingga 24 jam sebelum tag bertransisi ke `activated` status dan data menjadi kueri. Mer `EarliestDatapointTimestamp` upakan saat data tersedia, bukan saat tag diaktifkan.
- Batas kunci tag:  Setiap akun dapat memantau hingga lima kunci tag. Manager-provided Tag kapasitas tidak dihitung dalam batas ini.
- Persyaratan karakter kunci  tag: Kunci tag dapat berisi huruf Unicode, digit, spasi putih, dan karakter berikut:`_ . : / = + @ -`. Kunci tag tidak boleh melebihi 128 karakter.
- Re-activation:  Jika Anda menonaktifkan dan mengaktifkan kembali kunci tag yang sama, hanya data baru yang tersedia. `EarliestDatapointTimestamp`Reset dengan setiap aktivasi.

- Tag yang ditangguhkan: Setiap kunci tag mendukung hingga 100.000 nilai tag unik. Jika kunci tag melebihi ambang batas ini, kunci tersebut dipindahkan ke `suspended` status. Tag masih diperhitungkan dalam batas Anda tetapi data tidak lagi dicerna. Kurangi jumlah nilai unik untuk tag atau nonaktifkan untuk mengosongkan ruang untuk kunci tag lain.

## Menavigasi Manajer Kapasitas di AWS konsol

Konsol Capacity Manager diatur ke dalam tab yang menyediakan tampilan berbeda dari data kapasitas Anda:

- Dasbor — Menyediakan ikhtisar tingkat tinggi dari semua Reservasi On-Demand Kapasitas, On-Demand dan penggunaan Spot, termasuk metrik utama untuk membantu meningkatkan postur kapasitas Anda.
- Penggunaan — Menyediakan ikhtisar pola penggunaan instans untuk On-Demand dan instans Spot. Analisis cakupan berdasarkan Reservasi Kapasitas dan identifikasi peluang pengoptimalan melalui pengelompokan dan penyaringan yang fleksibel.
- Reservasi — Menyediakan analisis pemanfaatan Reservasi Kapasitas, kemampuan manajemen, dan metrik reservasi terperinci di seluruh akun dan Wilayah.
- Spot — Memantau pola penggunaan Spot dan menyediakan analisis biaya untuk instans Spot di seluruh akun dan Wilayah.
- Ekspor data — Mengelola konfigurasi ekspor data ke Amazon S3, termasuk penjadwalan, pemformatan, dan pemilihan template.
- Pengaturan — Menyediakan opsi konfigurasi layanan termasuk akses organisasi, pengaturan regional, dan manajemen kunci tag yang dipantau. Anda dapat mengaktifkan kunci tag dari sumber daya Amazon EC2 untuk digunakan sebagai dimensi saat mengelompokkan dan memfilter metrik kapasitas. Untuk informasi selengkapnya, lihat Mengelola kunci tag yang dipantau.

Dalam tab Penggunaan dan Reservasi, Manajer Kapasitas menyediakan struktur navigasi hierarkis yang memungkinkan Anda menelusuri dari ikhtisar tingkat tinggi hingga informasi sumber daya terperinci. Memahami pola navigasi ini membantu Anda menganalisis data kapasitas secara efisien dan mengidentifikasi peluang pengoptimalan.

Topik

- Hierarki navigasi
- Lihat rincian dan detail untuk Penggunaan dan Reservasi

## Hierarki navigasi

Manajer Kapasitas menggunakan struktur navigasi tiga tingkat untuk Penggunaan dan Reservasi:

1. Halaman ikhtis ar - High-level ringkasan dengan metrik agregat.
2. Halaman rincian - Analisis terperinci dengan opsi penyaringan dan pengelompokan
3. Halaman detail:
   - Detail penggunaan — Informasi tentang kombinasi dimensi yang Anda pilih, yang menyediakan statistik dan data deret waktu untuk membantu Anda memahami pola penggunaan.
   - Detail reservasi — Informasi tentang Reservasi Kapasitas tertentu termasuk statistik pemanfaatan, pola penggunaan dari waktu ke waktu, dan detail konfigurasi.

> **Note**
>
> Spot mengikuti struktur yang disederhanakan dengan hanya halaman ikhtisar.

## Lihat rincian dan detail untuk Penggunaan dan Reservasi

Tab Penggunaan dan Reservasi mengikuti struktur navigasi tiga tingkat yang sama, memungkinkan Anda untuk maju dari ikhtisar ke perincian ke halaman detail. Proses untuk mengakses halaman rincian dan detail serupa, dengan hanya sedikit perbedaan di mana tautan navigasi berada di dalam setiap tab.

Untuk mengakses halaman rincian sumber daya

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab untuk jenis sumber daya yang ingin Anda analisis: Penggunaan atau Reservasi.
4. Di bagian Agregasi, cari tautan rincian:
   - Penggunaan: Di kolom T indakan, pilih Lihat rincian.

> **Note**
>
> Bergantung pada ukuran layar atau jumlah dimensi yang telah Anda terapkan, Anda mungkin perlu menggulir secara horizontal melintasi halaman untuk menemukan tautan Lihat perincian.

- Reservasi: Di kolom Reservasi, pilih nomor (nomornya adalah tautan) reservasi yang ingin Anda lihat.

Untuk mengakses halaman detail

1. Pada halaman rincian, arahkan ke bagian yang relevan:
   - Penggunaan: Bagian kombinasi dimensi unik.
   - Reserv asi: Bagian Reservasi.
2. Di bagian yang relevan dari sumber daya yang Anda pilih, akses halaman detail.
   - Penggunaan: Di kolom Tindakan, pilih Lihat detail.

> **Note**
>
> Bergantung pada ukuran layar atau jumlah dimensi yang telah Anda terapkan, Anda mungkin perlu menggulir secara horizontal melintasi halaman untuk menemukan tautan Lihat detail.

   - Reservasi: Di kolom ID Reservasi, pilih reservasi yang ingin Anda lihat.

## Mengekspor data Capacity Manager Anda

Anda dapat mengekspor data kapasitas dari EC2 Capacity Manager ke Amazon S3 untuk mengaktifkan analisis lebih lanjut, membuat laporan khusus, atau mengintegrasikan dengan AWS layanan lain. Anda dapat mengekspor data Anda dalam format CSV atau Parquet. Di bagian berikut, Anda akan menemukan informasi tentang cara mengekspor data Manajer Kapasitas Anda.

> **Note**
>
> Capacity Manager hanya mengizinkan satu ekspor data per Akun AWS.

> **Note**
>
> Dalam kasus yang jarang terjadi bahwa ekspor data perlu didorong ulang karena masalah data, file baru akan mengganti file yang ada untuk jam itu.

Topik


- Menyiapkan bucket Amazon S3 untuk ekspor data Capacity Manager
- Membuat ekspor data untuk data Manajer Kapasitas


## Menyiapkan bucket Amazon S3 untuk ekspor data Capacity Manager

Untuk menerima ekspor data Capacity Manager, Anda harus memiliki bucket Amazon S3 Akun AWS untuk menerima dan menyimpan file ekspor Anda. Saat membuat ekspor data di konsol Capacity Manager, Anda dapat memilih bucket Amazon S3 yang sudah ada yang Anda miliki atau membuat bucket baru.

Dalam kedua kasus tersebut, Anda harus menerapkan kebijakan bucket yang diperlukan untuk mengizinkan Manajer Kapasitas mengirimkan file ekspor. Mengedit kebijakan ini di konsol Amazon S3 atau mengubah pemilik bucket setelah Anda membuat ekspor data akan mencegah Capacity Manager mengirimkan ekspor Anda.

Untuk membuat bucket Amazon S3, lihat Membuat bucket S3 di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Kebijakan berikut harus diterapkan ke bucket S3 agar Capacity Manager dapat mengirimkan ekspor data:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.capacitymanager.amazonaws.com"
            },
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "111122223333"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:ec2:us-east-1:111122223333:capacity-
manager-data-export/*"
                }
            }
        }
    ]
}
```

Kebijakan bucket ini membantu memastikan bahwa file ekspor data Manajer Kapasitas dapat dikirim dengan aman ke bucket Anda. Secara khusus:

- Setiap kali ekspor data Capacity Manager dikirimkan, AWS pertama-tama konfirmasikan apakah bucket masih dimiliki oleh akun yang mengatur ekspor. Jika kepemilikan bucket telah berubah, ekspor tidak akan dikirimkan. Ini membantu memastikan keamanan data Manajer Kapasitas. Kebijakan bucket ini memungkinkan AWS (`"Effect": "Allow"`) untuk memeriksa akun mana yang memiliki bucket (`"Action": ["s3:ListBucket"]`).
- Kebijakan ini memberikan izin kepada layanan Capacity Manager (`"Service": "ec2.capacitymanager.amazonaws.com"`) untuk menulis file ekspor (`"Action": "s3:PutObject"`) dan membaca objek (`"Action": "s3:GetObject"`) untuk menyalin data ke bucket Anda.

## Membuat ekspor data untuk data Manajer Kapasitas

Untuk membuat ekspor data, Anda dapat menggunakan halaman Ekspor Data di konsol Capacity Manager atau AWS CLI.

### Prasyarat

Anda harus membuat bucket Amazon Simple Storage Service (Amazon S3). Anda harus memastikan hal-hal berikut:

- Bucket S3 Anda harus berada di tempat yang sama Wilayah AWS di mana Anda mengaktifkan Manajer Kapasitas.
- Bucket S3 Anda memiliki kebijakan izin yang diperlukan untuk layanan Capacity Manager untuk mengirimkan file.

Untuk informasi selengkapnya, lihat Menyiapkan bucket Amazon S3 untuk ekspor data Capacity Manager.

### Prosedur

Anda dapat mengekspor data Manajer Kapasitas menggunakan AWS Konsol, AWS CLI, atau PowerShell.

#### Console

Untuk membuat ekspor data

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Manajer Kapasitas.
3. Pilih tab Ekspor data.
4. Pilih Buat ekspor data.
5. Konfigurasikan properti ekspor Anda, lokasi pengiriman, dan tag (opsional).
6. Pilih Buat.

#### AWS CLI

Untuk membuat ekspor data

Gunakan perintah berikut untuk membuat ekspor data dengan konfigurasi yang ditentukan:

```
aws ec2 create-capacity-manager-data-export \
    --s3-bucket-name my-exports-bucket \
    --s3-bucket-prefix capacity-data-exports \
    --schedule hourly \
    --output-format parquet/CSV \
    --tag-specifications 'ResourceType=capacity-manager-data-
export,Tags=[{Key=environment,Value=production}]'
```

> **Note**
>
> `--tag-specifications`Parameter dalam perintah di atas menerapkan tag sumber daya ke sumber daya ekspor data itu sendiri (misalnya, untuk alokasi biaya atau organisasi). Ini terpisah dari kunci tag yang dipantau, yang menentukan kolom tag yang disertakan dalam data yang diekspor. Untuk informasi selengkapnya tentang kunci tag yang dipantau, lihatMengelola kunci tag yang dipantau.

PowerShell

Untuk membuat ekspor data

Gunakan New-EC2CapacityManagerDataExport cmdlet.

```
New-EC2CapacityManagerDataExport `
    -S3BucketName "my-exports-bucket" `
    -S3BucketPrefix "capacity-data-exports" `
    -Schedule "hourly" `
    -OutputFormat "parquet" `
    -TagSpecification @([Amazon.EC2.Model.TagSpecification]@{
        ResourceType = "capacity-manager-data-export"
        Tags = @([Amazon.EC2.Model.Tag]@{
            Key   = "environment"
            Value = "production"
        })
    })
```

Menandai kolom dalam ekspor data

Ekspor data yang baru dibuat menyertakan nilai tag sebagai kolom tambahan untuk kunci tag yang dipantau dan Manager-provided tag Kapasitas yang diaktifkan. Untuk informasi selengkapnya, lihat Tag dalam ekspor data di Mengelola kunci tag yang dipantau.

> **Note**
>
> Jika Anda belum mengaktifkan kunci tag yang dipantau, ekspor Anda akan tetap menyertakan kolom untuk Manager-provided tag Kapasitas (misalnya,`aws:autoscaling:groupName`,`aws:eks:cluster-name`,`eks:kubernetes-node-pool-name`, dan`karpenter.sh/nodepool`).

# Mengelola driver perangkat untuk instans EC2

Driver perangkat adalah komponen perangkat lunak yang berkomunikasi dengan perangkat keras virtual untuk instans Amazon EC2 Anda. Untuk mencegah kesalahan sistem, masalah kinerja, dan perilaku tak terduga lainnya, penting untuk selalu memperbarui driver Anda. Itu terutama berlaku untuk driver yang dapat memiliki dampak kuat pada kinerja sistem tergantung pada penggunaan Anda, seperti jaringan, grafik, dan driver perangkat penyimpanan. Rilis driver baru dapat menyertakan perbaikan cacat atau memperkenalkan fungsionalitas yang diperluas yang mungkin ingin Anda manfaatkan untuk instance yang sedang berjalan.

## Driver jaringan

Distribusi Linux dapat menggabungkan fitur jaringan seperti Elastic Network Adapter (ENA) atau Elastic Fabric Adapter (EFA) dalam kernel. Namun, waktunya dapat bervariasi untuk implementasi fitur driver kernel dalam distribusi yang berbeda.

Driver kernel ENA dan EFA Linux tersedia dari GitHub repositori Amazon Drivers. Untuk informasi selengkapnya dan tautan ke driver yang tersedia, lihat Driver Amazon di GitHub.

Untuk informasi lebih lanjut tentang driver ENA, lihatAktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda. Untuk informasi selengkapnya tentang driver EFA, lihat Memulai topik di Adaptor Kain Elastis untuk AI/ML dan beban kerja HPC di Amazon EC2 bagian panduan ini.

Untuk menginstal atau memperbarui driver jaringan pada instance Windows, lihat topik berikut:

- Instal driver ENA di Windows
- Instal yang terbaru AWS Pengemudi PV

Untuk informasi selengkapnya, lihat Driver paravirtual untuk instans Windows.

> **Note**
>
> EFA tidak didukung pada instance Windows.

## Driver grafis

Untuk menginstal atau memperbarui driver grafis, lihat topik berikut:

- Driver AMD untuk instans EC2 Anda
- Driver NVIDIA untuk instans Amazon EC2 Anda

## Driver perangkat penyimpanan

Untuk menginstal atau memperbarui driver penyimpanan, lihat topik berikut:

- Untuk instance Linux, lihat Menginstal atau memutakhirkan driver NVMe di Panduan Pengguna Amazon EBS.
- Untuk instance Windows, lihatNVMe driver.

## Driver perangkat Windows lainnya

Untuk menginstal atau memperbarui driver perangkat Windows lainnya, lihat topik berikut:

- Driver perangkat Windows lainnya

## Driver AMD untuk instans EC2 Anda

Instans dengan GPU AMD terlampir, seperti instans G4ad, harus menginstal driver AMD yang sesuai. Tergantung pada kebutuhan Anda, Anda dapat menggunakan AMI dengan driver yang telah diinstal terlebih dahulu atau mengunduh driver dari Amazon S3.

Untuk menginstal driver NVIDIA pada instans dengan GPU NVIDIA terpasang, seperti instans G4dn, lihat Driver NVIDIA sebagai gantinya.

Daftar Isi


- Driver Perangkat Lunak AMD Radeon Pro untuk Korporasi
- AMI dengan driver AMD terinstal
- Mengunduh driver AMD


## Driver Perangkat Lunak AMD Radeon Pro untuk Korporasi

Driver AMD Radeon Pro Software for Enterprise dibangun untuk memberikan support untuk kasus penggunaan grafis kelas profesional. Dengan menggunakan driver tersebut, Anda dapat mengonfigurasi instans dengan dua layar 4K per GPU.

API yang didukung

- OpenGL, OpenCL
- Vulkan
- Advanced Media Framework AMD
- API Akselerasi Video
- DirectX 9 dan setelahnya
- Media Foundation Transform Perangkat Keras Microsoft

## AMI dengan driver AMD terinstal

AWS menawarkan Gambar Mesin Amazon (AMI) yang berbeda yang disertakan dengan driver AMD yang diinstal. Buka Penawaran Marketplace dengan driver AMD.

## Mengunduh driver AMD

Jika tidak menggunakan AMI dengan driver AMD terinstal, Anda dapat mengunduh driver AMD dan menginstalnya pada instans Anda. Hanya versi sistem operasi berikut yang mendukung driver AMD:

- Amazon Linux 2 dengan kernel versi 5.10
- Ubuntu 24.04

- Windows Server 2016
- Windows Server 2019
- Windows Server 2022

Unduhan ini hanya tersedia untuk AWS pelanggan. Dengan mengunduh driver ini, berarti Anda setuju untuk menggunakan perangkat lunak yang diunduh hanya guna mengembangkan AMI untuk digunakan dengan perangkat keras AMD Radeon Pro V520. Setelah menginstal perangkat lunak, Anda terikat oleh ketentuan Perjanjian Lisensi Pengguna Akhir AMD.

Instal driver AMD di instans Amazon Linux 2 Linux Anda

1. Hubungkan dengan instans Linux Anda.
2. Instal AWS CLI pada instance Linux Anda dan konfigurasikan kredenSIAL default. Untuk informasi selengkapnya, lihat Menginstal AWS CLI dalam Panduan Pengguna AWS Command Line Interface .

   > **Important**
   >
   > Pengguna atau peran Anda harus memiliki izin yang diberikan yang berisi AmazonS3ReadOnlyAccess kebijakan. Untuk informasi selengkapnya, lihat AWS kebijakan terkelola: AmazonS3ReadOnlyAccess di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

3. Inst gcc almake,dialog, dan, jika mereka belum diinstal.

   ```
   $ sudo yum install -y gcc make dialog
   ```

4. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

   ```
   $ sudo amazon-linux-extras install epel -y
   $ sudo yum update -y
   ```

5. Boot ulang instans.

   ```
   $ sudo reboot
   ```

6. Hubungkan kembali diri Anda ke instans setelah boot ulang.
7. Unduh driver AMD terbaru.

```
$ aws s3 cp --recursive s3://ec2-amd-linux-drivers/latest/ .
```

8. Ekstrak file.

```
$ tar -xzf amdgpu-pro-*rhel*.tar.gz
```

9. Instal paket amdgpu-install. Ganti *version* dengan nama direktori versi yang dibuat saat Anda mengekstrak driver.

```
$ cd rhel_7/version/x86_64/a/
$ sudo yum install -y ./amdgpu-install-*.el7.noarch.rpm
```

10. Ubah `/etc/yum.repos.d/amdgpu.repo`. Set `baseurl` el ke `x86_64` direktori yang diekstrak (misalnya,`baseurl=file:///home/ec2-user/rhel_7/version/x86_64/`). Atur `gpgcheck=0`.
11. Ubah `/etc/yum.repos.d/amdgpu-proprietary.repo`. Set `baseurl` el ke direktori yang sama. Atur `gpgcheck=0` dan`enabled=1`.
12. Ubah `/etc/yum.repos.d/rocm.repo`. Set `baseurl` el ke direktori yang sama. Atur `gpgcheck=0`.
13. Instal driver GPU AMD untuk Amazon Linux 2.

```
$ sudo amdgpu-install --usecase=workstation --vulkan=pro --accept-eula -y
```

14. Boot ulang instans.

```
$ sudo reboot
```

15. Konfirmasikan bahwa pengemudi berfungsi.

```
$ sudo dmesg | grep amdgpu
```

Responsnya akan terlihat seperti berikut:

```
Initialized amdgpu
```

Instal driver AMD pada instance Ubuntu Linux Anda

1. Hubungkan dengan instans Linux Anda.
2. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
$ sudo apt-get update -y && sudo apt-get upgrade -y
```

3. Instal firmware Linux dan modul kernel.

```
$ sudo apt install -y linux-firmware linux-modules-extra-$(uname -r)
```

4. Boot ulang instans.

```
$ sudo reboot
```

5. Hubungkan kembali diri Anda ke instans setelah boot ulang.
6. Unduh paket driver AMD terbaru untuk Ubuntu 24.04 dari Driver Linux® untuk AMD Radeon™ Graphics di situs web AMD. Dalam perintah berikut, ganti *version* dengan string versi dari nama file yang diunduh (misalnya,`31.30.313000-1_all`).

```
$ sudo apt install -y ./amdgpu-install_version.deb
```

7. Instal driver GPU AMD untuk Ubuntu.

```
$ sudo amdgpu-install --usecase=graphics --vulkan=radv -y
```

8. Boot ulang instans.

```
$ sudo reboot
```

9. Konfirmasikan bahwa pengemudi berfungsi.

```
$ sudo dmesg | grep amdgpu
```

Responsnya akan terlihat seperti berikut:

```
Initialized amdgpu
```

Instal driver AMD pada instance Windows Anda

1. Hubungkan ke instance Windows Anda dan buka PowerShell jendela.
2. Konfigurasikan kredenSIAL default untuk AWS Tools for Windows PowerShell pada instance Windows Anda. Untuk informasi selengkapnya, lihat Memulai AWS Tools for Windows PowerShell di Panduan Pengguna.Alat AWS untuk PowerShell

   > **Important**
   >
   > Pengguna atau peran Anda harus memiliki izin yang diberikan yang berisi AmazonS3ReadOnlyAccess kebijakan. Untuk informasi selengkapnya, lihat AWS kebijakan terkelola: AmazonS3ReadOnlyAccess di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

3. Atur awalan kunci sesuai dengan versi Windows Anda:

   - Windows 10 dan Windows 11

     ```
     $KeyPrefix = "latest/AMD_GPU_WINDOWS10"
     ```

   - Windows Server 2016

     ```
     $KeyPrefix = "archives"
     ```

   - Windows Server 2019

     ```
     $KeyPrefix = "latest/AMD_GPU_WINDOWS_2K19" # use "archives" for Windows Server
      2016
     ```

   - Windows Server 2022

     ```
     $KeyPrefix = "latest/AMD_GPU_WINDOWS_2K22"
     ```

4. Unduh driver dari Amazon S3 ke desktop Anda menggunakan PowerShell perintah berikut.

   ```
   $Bucket = "ec2-amd-windows-drivers"
   $LocalPath = "$home\Desktop\AMD"
   $Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
   foreach ($Object in $Objects) {
   $LocalFileName = $Object.Key
   if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
   ```

```
    $LocalFilePath = Join-Path $LocalPath $LocalFileName
    Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
    }
}
```

5. Buka zip file driver yang diunduh dan jalankan penginstal menggunakan PowerShell perintah berikut.

```
Expand-Archive $LocalFilePath -DestinationPath "$home\Desktop\AMD\$KeyPrefix" -
Verbose
```

Sekarang, periksa isi direktori baru. Nama direktori dapat diambil menggunakan `Get-ChildItem` PowerShell perintah.

```
Get-ChildItem "$home\Desktop\AMD\$KeyPrefix"
```

Output harus serupa dengan yang berikut ini:

```
Directory: C:\Users\Administrator\Desktop\AMD\latest

Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----       10/13/2021  12:52 AM                210414a-365562C-Retail_End_User.2
```

Instal driver:

```
pnputil /add-driver $home\Desktop\AMD\$KeyPrefix\*.inf /install /subdirs
```

6. Ikuti petunjuk untuk menginstal driver dan mem-boot ulang instans Anda sesuai kebutuhan.
7. Untuk memverifikasi bahwa GPU berfungsi dengan benar, periksa Device Manager. Anda akan melihat “AMD Radeon Pro V520 MxGPU” terdaftar sebagai adaptor tampilan.
8. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, siapkan protokol tampilan berkinerja tinggi, Amazon DCV.

# Driver NVIDIA untuk instans Amazon EC2 Anda

Instans dengan GPU NVIDIA terlampir, seperti tipe instance seri P atau G, harus menginstal driver NVIDIA yang sesuai. Bergantung pada jenis instansnya, Anda dapat mengunduh driver NVIDIA publik, mengunduh driver dari Amazon S3 yang hanya tersedia untuk AWS pelanggan, atau menggunakan AWS AMI dengan driver yang sudah diinstal sebelumnya.

Untuk menginstal driver AMD pada instans dengan GPU AMD terpasang, seperti instans G4ad, lihat Driver AMD sebagai gantinya.

Topik


- Jenis driver NVIDIA
- Driver yang tersedia berdasarkan tipe instans
- Opsi penginstalan
- Gunakan AMI yang menyertakan driver NVIDIA
- Instal driver publik NVIDIA
- Instal driver NVIDIA GRID (instans G7e, G6, Gr6, G6e, G6f, GR6f, G5, G4dn, dan G3)
- Instal driver gaming NVIDIA (instans G7e, G6, G6e, G5, dan G4dn)


## Jenis driver NVIDIA

Berikut ini adalah jenis utama driver NVIDIA yang dapat digunakan dengan GPU-based instance.

Driver Tesla

Driver ini ditujukan terutama untuk beban kerja komputasi, yang menggunakan GPU untuk tugas komputasi seperti penghitungan floating-point paralel untuk machine learning dan transformasi Fourier cepat untuk aplikasi komputasi performa tinggi.

Driver GRID

Driver ini disertifikasi untuk memberikan performa optimal untuk aplikasi visualisasi profesional yang melakukan render konten seperti model 3D atau video resolusi tinggi. Anda dapat mengonfigurasi driver GRID untuk mendukung dua mode. Quadro Virtual Workstations menyediakan akses ke empat layar 4K per GPU. GRID vApps menyediakan kemampuan hosting Aplikasi RDSH.

Driver game

Driver ini berisi optimisasi untuk game dan diperbarui secara frekuen untuk memberikan peningkatan performa. Driver ini juga mendukung satu layar 4K per GPU.

Mode terkonfigurasi

Di Windows, driver Tesla dikonfigurasi untuk berjalan dalam mode Tesla Compute Cluster (TCC). Driver GRID dan game dikonfigurasi untuk berjalan dalam mode Windows Display Driver Model (WDDM). Dalam mode TCC, kartu tersebut dikhususkan untuk beban kerja komputasi. Dalam mode WDDM, kartu mendukung beban kerja komputasi dan grafis.

Panel kontrol NVIDIA

Panel kontrol NVIDIA didukung dengan driver GRID dan Gaming. Panel kontrol NVIDIA tidak didukung dengan driver Tesla.

API yang didukung untuk Tesla, GRID, dan driver gaming

- OpenCL, OpenGL, dan Vulkan
- NVIDIA CUDA dan pustaka terkait (misalnya, cuDNN, TensorRT, nvJPEG, and cuBLAS)
- NVENC untuk enkode video dan NVDEC untuk dekode video
- Windows-only API: DirectX, Direct2D, Akselerasi Video DirectX, Raytracing DirectX

## Driver yang tersedia berdasarkan tipe instans

Tabel berikut merangkum driver NVIDIA yang didukung untuk setiap tipe instans GPU.

| Jenis instans | Driver Tesla | Driver GRID | Driver game |
|---|---|---|---|
| G3 | Ya | Ya | Tidak |
| G4dn | Ya | Ya | Ya |
| G5 | Ya | Ya | Ya |
| G5g | Ya [1] | Tidak | Tidak |
| G6 | Ya | Ya | Ya |

| Jenis instans | Driver Tesla | Driver GRID | Driver game |
|---|---|---|---|
| G6e | Ya | Ya | Ya |
| G6f | Tidak | Ya | Tidak |
| Gr 6 | Ya | Ya | Tidak |
| Gr6f | Tidak | Ya | Tidak |
| G7e | Ya | Ya | Ya |
| P3 | Ya | Tidak | Tidak |
| P4d | Ya | Tidak | Tidak |
| P4de | Ya | Tidak | Tidak |
| P5 | Ya | Tidak | Tidak |
| P5e | Ya | Tidak | Tidak |
| P5en | Ya | Tidak | Tidak |
| P6-B200 | Ya | Tidak | Tidak |
| P6e-GB200 | Ya | Tidak | Tidak |
| P6-B300 | Ya | Tidak | Tidak |

[1] Driver Tesla ini juga mendukung aplikasi grafis yang dioptimalkan khusus untuk platform ARM64

## Opsi penginstalan

Gunakan salah satu opsi berikut untuk mendapatkan driver NVIDIA yang diperlukan untuk instans GPU Anda.

Pilihan

1. Gunakan AMI yang menyertakan driver NVIDIA
2. Instal driver publik NVIDIA

3. Instal driver NVIDIA GRID (instans G7e, G6, Gr6, G6e, G6f, GR6f, G5, G4dn, dan G3)
4. Instal driver gaming NVIDIA (instans G7e, G6, G6e, G5, dan G4dn)

## Gunakan AMI yang menyertakan driver NVIDIA

AWS dan NVIDIA menawarkan Amazon Machine Images (AMI) yang berbeda yang disertakan dengan driver NVIDIA yang diinstal.

- Penawaran pasar dengan pengemudi umum Tesla
- Penawaran pasar dengan driver GRID
- Penawaran pasar dengan driver Gaming

Untuk meninjau pertimbangan yang bergantung pada platform sistem operasi (OS) Anda, pilih tab yang berlaku untuk AMI Anda.

Linux

Untuk memperbarui versi driver yang diinstal menggunakan salah satu dari AMI ini, Anda harus menghapus instalasi paket NVIDIA dari instans untuk menghindari konflik versi. Gunakan perintah ini untuk menghapus paket NVIDIA:

```
[ec2-user ~]$ sudo yum erase nvidia cuda
```

Paket kit alat CUDA memiliki dependensi terhadap driver NVIDIA. Menghapus instalasi paket NVIDIA akan menghapus kit alat CUDA. Anda harus menginstal ulang kit alat CUDA setelah menginstal driver NVIDIA.

Windows

Jika Anda membuat AMI Windows kustom menggunakan salah satu AWS Marketplace penawaran, AMI harus berupa gambar standar yang dibuat dengan Windows Sysprep untuk memastikan bahwa driver GRID berfungsi. Lihat informasi yang lebih lengkap di Buat AMI Amazon EC2 menggunakan Windows Sysprep.

## Instal driver publik NVIDIA

Jika AWS Marketplace AMI yang dijelaskan dalam Gunakan AMI yang menyertakan driver NVIDIA tidak sesuai dengan kasus penggunaan Anda, Anda dapat menginstal driver publik dan membawa lisensi Anda sendiri. Opsi instalasi meliputi yang berikut:

- Opsi 1: Driver-only instal
- Opsi 2: Instal dengan toolkit CUDA(direkomendasikan untuk distribusi Linux)

P6-B200 dan pertimbangan jenis P6-B300 instance

P6-B300 Platform P6-B200 dan unik karena mengekspos kartu antarmuka jaringan (NIC) Mellanox ConnectX ke instance sebagai perangkat PCIe. NIC ini tidak bertindak sebagai antarmuka jaringan tipikal tetapi berfungsi sebagai jembatan NVSwitch yang menyediakan jalur kontrol untuk menginisialisasi dan mengkonfigurasi NVFabric, yang merupakan topologi NVLink dari interkoneksi GPU.

Untuk menginisialisasi sistem sepenuhnya, NVIDIA Fabric Manager harus mengkonfigurasi `NVFabric` dan menetapkan topologi NVSwitch. Hal ini memungkinkan modul InfiniBand kernel untuk berkomunikasi dengan NIC Mellanox ConnectX.

NVIDIA Fabric Manager termasuk dalam toolkit CUDA. Kami merekomendasikan Opsi 2: Instal dengan toolkit CUDA untuk jenis contoh ini.

Opsi 1: Driver-only instal

Untuk menginstal driver tertentu, masuk ke instance Anda dan unduh driver publik NVIDIA 64-bit untuk jenis instans dari http://www.nvidia.com/Download/Find.aspx. Untuk Jenis Produk, Seri Produk, dan Produk, gunakan opsi yang ditunjukkan pada tabel berikut.

Kemudian ikuti petunjuk Instal asi Repositori Lokal di Panduan Instalasi Driver NVIDIA.

> **Note**
>
> P6-B200 dan jenis P6-B300 instance memerlukan instalasi dan konfigurasi paket tambahan yang disertakan dengan NVIDIA CUDA Toolkit. Untuk informasi selengkapnya, lihat instruksi untuk distribusi Linux Anda diOpsi 2: Instal dengan toolkit CUDA.

| Instans | Jenis produk | Seri produk | Produk | Versi driver minimum |
|---|---|---|---|---|
| G3 | Tesla | M-Class | M60 | -- |
| G4dn | Tesla | T-Series | T4 | -- |
| G5 | Tesla | A-Series | A10 | 470,00 atau lebih baru |
| G5g1 | Tesla | T-Series | T4G | 470.82.01 atau yang lebih baru |
| G6 | Tesla | L-Series | L4 | 525.0 atau yang lebih baru |
| G6e | Tesla | L-Series | L40S | 535.0 atau yang lebih baru |
| Gr 6 | Tesla | L-Series | L4 | 525.0 atau yang lebih baru |
| G7 | Tesla | Seri RTX | RTX PRO 4500 Blackwell | 595 atau yang lebih baru |
| G7e | Tesla | Seri RTX | RTX PRO 6000 Blackwell | 575.0 atau yang lebih baru |
| P3 | Tesla | V-Series | V100 | -- |
| P4d | Tesla | A-Series | A100 | -- |
| P4de | Tesla | A-Series | A100 | -- |
| P5 | Tesla | H-Series | H100 | 530 atau yang lebih baru |
| P5e | Tesla | H-Series | H200 | 550 atau yang lebih baru |

| Instans | Jenis produk | Seri produk | Produk | Versi driver minimum |
|---|---|---|---|---|
| P5en | Tesla | H-Series | H200 | 550 atau yang lebih baru |
| P6-B200[2] | Tesla | HGX-Series | B200 | 570 atau yang lebih baru |
| P6e-GB200 | Tesla | HGX-Series | B200 | 570 atau yang lebih baru |
| P6-B300[2] | Tesla | HGX-Series | B300 | 580 atau yang lebih baru |

[1] Sistem operasi untuk instans G5g adalah Linux aarch64.

[2] Untuk jenis P6-B200 dan P6-B300 contoh, ada persyaratan penginstalan tambahan untuk mengkonfigurasi NVIDIA Fabric Manager.

Opsi 2: Instal dengan toolkit CUDA

Instruksi penginstalan sedikit berbeda menurut sistem operasi. Untuk menginstal driver publik pada instans Anda dengan toolkit NVIDIA CUDA, ikuti petunjuk untuk sistem operasi instans Anda. Misalnya sistem operasi yang tidak ditampilkan di sini, ikuti petunjuk untuk sistem operasi dan arsitektur jenis instans Anda di situs web Pengembang NVIDIA. Untuk informasi selengkapnya, lihat Unduhan https://developer.nvidia.com/cuda-downloads CUDA Toolkit.

Misalnya arsitektur tipe atau spesifikasi lainnya, lihat Spesifikasi komputasi yang dipercepat di referensi Jenis Instans Amazon EC2.

Amazon Linux 2023

Bagian ini mencakup instalasi toolkit NVIDIA CUDA pada instans Amazon Linux 2023. Contoh perintah di bagian ini didasarkan pada `x86_64` arsitektur.

Untuk `arm64-sbsa` perintah, lihat Unduhan CUDA Toolkit dan pilih opsi yang berlaku untuk distribusi Anda. Instruksi muncul setelah Anda membuat pilihan akhir Anda.

Prasyarat

Sebelum menginstal toolkit dan driver, jalankan perintah berikut untuk memastikan bahwa Anda memiliki versi header kernel dan paket pengembangan yang benar.

```
[ec2-user ~]$ sudo dnf install kernel-devel-$(uname -r) kernel-headers-$(uname -r) -y
```

Unduh toolkit dan driver

Pilih jenis instalasi yang akan digunakan untuk instance Anda, dan ikuti langkah-langkah terkait.

AL2023 NVIDIA repository

Amazon Linux 2023 menyediakan driver GPU NVIDIA dan paket toolkit CUDA melalui repositori khusus yang AWS memelihara. AWS memenuhi syarat repositori ini dengan kandidat rilis AL2023 dan memberikan saran keamanan melalui Pusat Keamanan Amazon Linux. Kami merekomendasikan opsi ini untuk instans AL2023 karena menyederhanakan penginstalan dan terus memperbarui driver melalui al `dnf update` ur kerja standar.

Untuk petunjuk, lihat driver NVIDIA di Panduan Pengguna Amazon Linux 2023.

RPM local installation

Anda dapat mengikuti petunjuk ini untuk mengunduh bundel repositori penginstal toolkit CUDA ke instance Anda, lalu ekstrak dan mendaftarkan bundel yang ditentukan.

Untuk melihat petunjuk di situs web pengembang NVIDIA, lihat Unduhan https://developer.nvidia.com/cuda-downloads?target_os=Linux&target_arch=x86_64&Distribution=Amazon-Linux&target_version=2023&target_type=rpm_local CUDA Toolkit.

```
[ec2-user ~]$ wget https://developer.download.nvidia.com/compute/cuda/13.0.0/
local_installers/cuda-repo-amzn2023-13-0-local-13.0.0_580.65.06-1.x86_64.rpm
[ec2-user ~]$ sudo rpm -i cuda-repo-amzn2023-13-0-
local-13.0.0_580.65.06-1.x86_64.rpm
```

RPM network installation

Anda dapat mengikuti petunjuk ini untuk mendaftarkan repositori CUDA dengan manajer paket pada instance Anda. Saat Anda menjalankan langkah-langkah penginstalan, manajer paket hanya mengunduh paket yang diperlukan.

Untuk melihat petunjuk di situs web pengembang NVIDIA, lihat Unduhan https://developer.nvidia.com/cuda-downloads?

target_os=Linux&target_arch=x86_64&Distribution=Amazon-Linux&target_version=2023&target_type=rpm_network CUDA Toolkit.

```
[ec2-user ~]$ sudo dnf config-manager --add-repo https://
developer.download.nvidia.com/compute/cuda/repos/amzn2023/x86_64/cuda-amzn2023.repo
```

Langkah-langkah yang tersisa sama untuk instalasi jaringan lokal RPM dan RPM.

1. Selesaikan instalasi toolkit CUDA

```
[ec2-user ~]$ sudo dnf clean all
[ec2-user ~]$ sudo dnf install cuda-toolkit -y
```

2. Instal varian modul kernel terbuka dari driver

```
[ec2-user ~]$ sudo dnf module install nvidia-driver:open-dkms -y
```

3. Instal Penyimpanan GPUDirect dan Manajer Fabric

```
[ec2-user ~]$ sudo dnf install nvidia-gds -y
[ec2-user ~]$ sudo dnf install nvidia-fabric-manager -y
```

4. Aktifkan Fabric Manager dan ketekunan driver

```
[ec2-user ~]$ sudo systemctl enable nvidia-fabricmanager
[ec2-user ~]$ sudo systemctl enable nvidia-persistenced
```

5. (P6-B200 dan P6-B300 hanya) Jenis instance ini memerlukan instalasi dan konfigurasi paket tambahan yang disertakan dengan NVIDIA CUDA Toolkit.

   1. Instal NVIDIA Link Subnet Manager dan`ibstat`.

      ```
      [ec2-user ~]$ sudo dnf install nvlink5
      ```

   2. Aktifkan pemuatan otomatis modul Infiniband saat startup.

      ```
      [ec2-user ~]$ echo "ib_umad" | sudo tee -a /etc/modules-load.d/modules.conf
      ```

6. Nyalakan ulang instance

```
[ec2-user ~]$ sudo reboot
```

Ubuntu 24.04

Bagian ini mencakup instalasi toolkit NVIDIA CUDA pada instance Ubuntu 24.04. Contoh perintah di bagian ini didasarkan pada `x86_64` arsitektur.

Untuk `arm64-sbsa` perintah, lihat Unduhan CUDA Toolkit dan pilih opsi yang berlaku untuk distribusi Anda. Instruksi muncul setelah Anda membuat pilihan akhir Anda.

Prasyarat

Sebelum menginstal toolkit dan driver, jalankan perintah berikut untuk memastikan bahwa Anda memiliki versi header kernel dan paket pengembangan yang benar.

```
$ apt install linux-headers-$(uname -r)
```

Unduh toolkit dan driver

Pilih jenis instalasi yang akan digunakan untuk instance Anda, dan ikuti langkah-langkah terkait.

Deb local installation

Anda dapat mengikuti petunjuk ini untuk mengunduh bundel repositori penginstal toolkit CUDA ke instance Anda, lalu ekstrak dan daftarkan bundel yang ditentukan.

Untuk melihat petunjuk di situs web pengembang NVIDIA, lihat Unduhan https://developer.nvidia.com/cuda-downloads?target_os=Linux&target_arch=x86_64&Distribution=Ubuntu&target_version=24.04&target_type=deb_local CUDA Toolkit.

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2404/
x86_64/cuda-ubuntu2404.pin
$ sudo mv cuda-ubuntu2404.pin /etc/apt/preferences.d/cuda-repository-pin-600
$ wget https://developer.download.nvidia.com/compute/cuda/13.0.0/
local_installers/cuda-repo-ubuntu2404-13-0-local_13.0.0-580.65.06-1_amd64.deb
$ sudo dpkg -i cuda-repo-ubuntu2404-13-0-local_13.0.0-580.65.06-1_amd64.deb
$ sudo cp /var/cuda-repo-ubuntu2404-13-0-local/cuda-*-keyring.gpg /usr/share/
keyrings/
```

Deb network installation

Anda dapat mengikuti petunjuk ini untuk mendaftarkan repositori CUDA dengan manajer paket pada instance Anda. Saat Anda menjalankan langkah-langkah penginstalan, manajer paket hanya mengunduh paket yang diperlukan.

Untuk melihat petunjuk di situs web pengembang NVIDIA, lihat Unduhan https://developer.nvidia.com/cuda-downloads?target_os=Linux&target_arch=x86_64&Distribution=Ubuntu&target_version=24.04&target_type=deb_netw CUDA Toolkit.

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2404/x86_64/
cuda-keyring_1.1-1_all.deb
$ sudo dpkg -i cuda-keyring_1.1-1_all.deb
```

Langkah-langkah yang tersisa sama untuk instalasi lokal dan jaringan.

1. Selesaikan instalasi toolkit CUDA

```
$ sudo apt update
$ sudo apt install cuda-toolkit -y
```

2. Instal varian modul kernel terbuka dari driver

```
$ sudo apt install nvidia-open -y
```

3. Instal Penyimpanan GPUDirect dan Manajer Fabric

```
$ sudo apt install nvidia-gds -y
$ sudo apt install nvidia-fabricmanager -y
```

4. Aktifkan Fabric Manager dan ketekunan driver

```
$ sudo systemctl enable nvidia-fabricmanager
$ sudo systemctl enable nvidia-persistenced
```

5. (P6-B200 dan P6-B300 hanya) Jenis instance ini memerlukan instalasi dan konfigurasi paket tambahan yang disertakan dengan NVIDIA CUDA Toolkit.

    1. Instal driver InfiniBand-specific perangkat terbaru dan utilitas diagnostik.

```
$ sudo apt install linux-modules-extra-$(uname -r) -y
$ sudo apt install infiniband-diags -y
```

2. Instal NVIDIA Link Subnet Manager.

```
$ sudo apt install nvlsm -y
```

6. Nyalakan ulang instance

```
sudo reboot
```

7. Perbarui jalur Anda dan tambahkan variabel lingkungan berikut.

```
$ export PATH=${PATH}:/usr/local/cuda-13.0/bin
$ export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/usr/local/cuda-13.0/lib64
```

Sistem operasi Windows

Untuk menginstal driver NVIDIA di Windows, ikuti langkah-langkah ini:

1. Buka folder tempat Anda mengunduh driver dan luncurkan file instalasi. Ikuti petunjuk untuk menginstal driver dan mem-boot ulang instans Anda sesuai kebutuhan.
2. Nonaktifkan adaptor tampilan bernama Microsoft Basic Display Adapter yang ditandai dengan ikon peringatan menggunakan Device Manager. Instal fitur Windows ini: Media Foundation dan Quality Windows Audio Video Experience.

> **Important**
>
> Jangan nonaktifkan adaptor tampilan bernama Microsoft Remote Display Adapter. Jika Microsoft Remote Display Adapter dinonaktifkan, koneksi Anda mungkin terputus dan upaya untuk menyambung ke instans setelah reboot mungkin gagal.

3. Periksa Manajer Perangkat untuk memverifikasi bahwa GPU berfungsi dengan benar.
4. Untuk mencapai kinerja terbaik dari GPU Anda, selesaikan langkah-langkah pengoptimalan di Optimalkan pengaturan GPU pada instans Amazon EC2.

## Instal driver NVIDIA GRID (instans G7e, G6, Gr6, G6e, G6f, GR6f, G5, G4dn, dan G3)

Unduhan ini hanya tersedia untuk AWS pelanggan. Dengan mengunduh, untuk mematuhi persyaratan AWS solusi yang disebutkan dalam Perjanjian Lisensi Pengguna Akhir NVIDIA GRID Cloud (EULA), Anda setuju untuk menggunakan perangkat lunak yang diunduh hanya untuk mengembangkan AMI untuk digunakan dengan perangkat keras NVIDIA L4, NVIDIA L40S, NVIDIA A10G, NVIDIA Tesla T4, atau NVIDIA Tesla M60. Anda dapat menggunakan driver GRID untuk membuat dan menggunakan AMI dalam AWS lingkungan. Setelah menginstal perangkat lunak, Anda terikat oleh persyaratan Perjanjian Lisensi Pengguna Akhir Cloud NVIDIA GRID. Untuk informasi tentang versi driver NVIDIA GRID untuk sistem operasi Anda, lihat Perangkat Lunak NVIDIA Virtual GPU (vGPU) di situs web NVIDIA.

Pertimbangan-pertimbangan

- Instans G7e memerlukan GRID 19.1 atau yang lebih baru untuk Linux dan 19.4 (582.16) atau yang lebih baru untuk Windows.
- Instans G6f dan Gr6f memerlukan GRID 18.4 hingga GRID 19.5.
- Instans G6e memerlukan GRID 17.4 atau yang lebih baru.
- Instans G6 dan Gr6 memerlukan GRID 17.1 atau yang lebih baru.
- Instans G5 memerlukan GRID 13.1 atau setelahnya (atau GRID 12.4 atau setelahnya).
- Instans G3 memerlukan resolusi DNS yang AWS disediakan agar lisensi GRID berfungsi.
- IMDSv2 hanya didukung dengan driver NVIDIA versi 14.0 atau lebih tinggi.
- Untuk instans Windows, jika Anda meluncurkan instance dari AMI Windows kustom, AMI harus berupa gambar standar yang dibuat dengan Windows Sysprep untuk memastikan bahwa driver GRID berfungsi. Untuk informasi selengkapnya, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.
- GRID 17.0 dan yang lebih baru tidak mendukung Windows Server 2019.
- GRID 14.2 dan yang lebih baru tidak mendukung Windows Server 2016.
- GRID 17.0 dan yang lebih baru tidak didukung dengan instans G3.
- Untuk instance Linux, Anda mungkin perlu menginstal atau memperbarui paket, seperti gcc, jika penginstal NVIDIA gagal dengan pesan kesalahan. Spesifikasinya tergantung pada versi sistem operasi dan kernel. Untuk informasi selengkapnya, lihat Portal Dukungan Perusahaan NVIDIA.

Prasyarat

- (Linux) Verifikasi bahwa AWS CLI diinstal pada instans Anda dan dikonfigurasi dengan kredenSIAL default. Untuk informasi selengkapnya, lihat Menginstal AWS CLI dalam Panduan Pengguna AWS Command Line Interface .
- (Windows) Konfigurasikan kredenSIAL default untuk AWS Tools for Windows PowerShell pada instans Anda. Untuk informasi selengkapnya, lihat Memulai AWS Tools for Windows PowerShell dalam Panduan Pengguna Alat AWS untuk PowerShell .
- Pengguna atau peran Anda harus memiliki izin yang diberikan yang berisi AmazonS3ReadOnlyAccess kebijakan.

Amazon Linux 2023

Untuk menginstal driver NVIDIA GRID pada instans Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo dnf update -y
```

2. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo dnf install gcc make
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel.

```
[ec2-user ~]$ sudo dnf install -y kernel-devel kernel-modules-extra
```

6. Unduh utilitas instalasi driver GRID menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

Beberapa versi driver GRID disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut.

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

7. Tambahkan izin untuk menjalankan utilitas penginstalan driver menggunakan perintah berikut.

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

8. Jalankan skrip instalasi mandiri sebagai berikut untuk menginstal driver GRID yang Anda unduh. Misalnya:

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

   Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

9. Konfirmasikan bahwa pengemudi berfungsi. Respons untuk perintah berikut mencantumkan versi driver NVIDIA yang diinstal dan detail tentang GPU.

```
[ec2-user ~]$ nvidia-smi -q | head
```

10. Jika Anda menggunakan perangkat lunak NVIDIA vGPU versi 14.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

11. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

12. (Opsional) Bergantung pada kasus penggunaan Anda, Anda dapat menyelesaikan langkah opsional berikut. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah-langkah ini.

    a. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

b. Mode NVIDIA Quadro Virtual Workstation diaktifkan secara default. Untuk mengaktifkan Aplikasi Virtual GRID untuk kemampuan hosting Aplikasi RDSH, selesaikan langkah-langkah aktivasi Aplikasi Virtual GRID di Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda.

Amazon Linux 2

Untuk menginstal driver NVIDIA GRID pada instans Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

2. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install gcc make
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel untuk versi kernel yang sedang berjalan.

```
[ec2-user ~]$ sudo yum install -y kernel-devel-$(uname -r)
```

6. Unduh utilitas instalasi driver GRID menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

Beberapa versi driver GRID disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut.

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

7. Tambahkan izin untuk menjalankan utilitas penginstalan driver menggunakan perintah berikut.

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

8. Jalankan skrip instalasi mandiri sebagai berikut untuk menginstal driver GRID yang Anda unduh. Misalnya:

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

Jika Anda menggunakan Amazon Linux 2 dengan kernel versi 5.10, gunakan perintah berikut untuk menginstal driver GRID.

```
[ec2-user ~]$ sudo CC=/usr/bin/gcc10-cc ./NVIDIA-Linux-x86_64*.run
```

Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

9. Konfirmasikan bahwa pengemudi berfungsi. Respons untuk perintah berikut mencantumkan versi driver NVIDIA yang diinstal dan detail tentang GPU.

```
[ec2-user ~]$ nvidia-smi -q | head
```

10. Jika Anda menggunakan perangkat lunak NVIDIA vGPU versi 14.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

11. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

12. (Opsional) Bergantung pada kasus penggunaan Anda, Anda dapat menyelesaikan langkah opsional berikut. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah-langkah ini.

    a. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

    b. Mode NVIDIA Quadro Virtual Workstation diaktifkan secara default. Untuk mengaktifkan Aplikasi Virtual GRID untuk kemampuan hosting Aplikasi RDSH, selesaikan langkah-

langkah aktivasi Aplikasi Virtual GRID di Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda.

CentOS 7 dan Red Hat Enterprise Linux 7

Untuk menginstal driver NVIDIA GRID pada instans Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

2. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install -y gcc make
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel untuk versi kernel yang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y kernel-devel-$(uname -r)
```

6. Nonaktifkan driver `nouveau` sumber terbuka untuk kartu grafis NVIDIA.

   a. Menambahkan `nouveau` ke file daftar hitam `/etc/modprobe.d/blacklist.conf`. Salin blok kode berikut dan tempelkan ke terminal.

```
[ec2-user ~]$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

   b. Edit file `/etc/default/grub` dan tambahkan baris berikut:

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

c. Bangun kembali konfigurasi Grub.

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

7. Unduh utilitas instalasi driver GRID menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

Beberapa versi driver GRID disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut.

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

8. Tambahkan izin untuk menjalankan utilitas penginstalan driver menggunakan perintah berikut.

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

9. Jalankan skrip instalasi mandiri sebagai berikut untuk menginstal driver GRID yang Anda unduh. Misalnya:

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

10. Konfirmasikan bahwa pengemudi berfungsi. Respons untuk perintah berikut mencantumkan versi driver NVIDIA yang diinstal dan detail tentang GPU.

```
[ec2-user ~]$ nvidia-smi -q | head
```

11. Jika Anda menggunakan perangkat lunak NVIDIA vGPU versi 14.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

12. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

13. (Opsional) Bergantung pada kasus penggunaan Anda, Anda dapat menyelesaikan langkah opsional berikut. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah-langkah ini.

    a. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

    b. Mode NVIDIA Quadro Virtual Workstation diaktifkan secara default. Untuk mengaktifkan Aplikasi Virtual GRID untuk kemampuan hosting Aplikasi RDSH, selesaikan langkah-langkah aktivasi Aplikasi Virtual GRID di Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda.

    c. Instal desktop/workstation paket GUI.

    ```
    [ec2-user ~]$ sudo yum groupinstall -y "Server with GUI"
    ```

CentOS Stream 8 dan Red Hat Enterprise Linux 8

Untuk menginstal driver NVIDIA GRID pada instans Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

2. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install -y gcc make
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel untuk versi kernel yang Anda jalankan.

```
[ec2-user ~]$ sudo dnf install -y elfutils-libelf-devel libglvnd-devel kernel-
devel-$(uname -r)
```

6. Unduh utilitas instalasi driver GRID menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

   Beberapa versi driver GRID disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut.

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

7. Tambahkan izin untuk menjalankan utilitas penginstalan driver menggunakan perintah berikut.

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

8. Jalankan skrip instalasi mandiri sebagai berikut untuk menginstal driver GRID yang Anda unduh. Misalnya:

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

   Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

9. Konfirmasikan bahwa pengemudi berfungsi. Respons untuk perintah berikut mencantumkan versi driver NVIDIA yang diinstal dan detail tentang GPU.

```
[ec2-user ~]$ nvidia-smi -q | head
```

10. Jika Anda menggunakan perangkat lunak NVIDIA vGPU versi 14.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

11. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

12. (Opsional) Bergantung pada kasus penggunaan Anda, Anda dapat menyelesaikan langkah opsional berikut. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah-langkah ini.

    a. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

    b. Mode NVIDIA Quadro Virtual Workstation diaktifkan secara default. Untuk mengaktifkan Aplikasi Virtual GRID untuk kemampuan hosting Aplikasi RDSH, selesaikan langkah-langkah aktivasi Aplikasi Virtual GRID di Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda.

    c. Instal paket workstation GUI.

    ```
    [ec2-user ~]$ sudo dnf groupinstall -y workstation
    ```

Rocky Linux 8

Untuk menginstal driver NVIDIA GRID pada instans Linux Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

2. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install -y gcc make
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel untuk versi kernel yang Anda jalankan.

```
[ec2-user ~]$ sudo dnf install -y elfutils-libelf-devel libglvnd-devel kernel-
devel-$(uname -r)
```

6. Unduh utilitas instalasi driver GRID menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

   Beberapa versi driver GRID disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut.

```
[ec2-user ~]$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

7. Tambahkan izin untuk menjalankan utilitas penginstalan driver menggunakan perintah berikut.

```
[ec2-user ~]$ chmod +x NVIDIA-Linux-x86_64*.run
```

8. Jalankan skrip instalasi mandiri sebagai berikut untuk menginstal driver GRID yang Anda unduh. Misalnya:

```
[ec2-user ~]$ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
```

   Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

9. Konfirmasikan bahwa pengemudi berfungsi. Respons untuk perintah berikut mencantumkan versi driver NVIDIA yang diinstal dan detail tentang GPU.

```
[ec2-user ~]$ nvidia-smi -q | head
```

10. Jika Anda menggunakan perangkat lunak NVIDIA vGPU versi 14.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

11. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

12. (Opsional) Bergantung pada kasus penggunaan Anda, Anda dapat menyelesaikan langkah opsional berikut. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah-langkah ini.

    a. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

    b. Mode NVIDIA Quadro Virtual Workstation diaktifkan secara default. Untuk mengaktifkan Aplikasi Virtual GRID untuk kemampuan hosting Aplikasi RDSH, selesaikan langkah-langkah aktivasi Aplikasi Virtual GRID di Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda.

Ubuntu dan Debian

Untuk menginstal driver NVIDIA GRID pada instans Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
$ sudo apt-get update -y
```

2. Install gcc dan make, jika belum terinstal.

```
$ sudo apt-get install -y gcc make
```

3. (Ubuntu) Mutakhirkan paket `linux-aws` untuk menerima versi terbaru.

```
$ sudo apt-get upgrade -y linux-aws
```

   (Debian) Mutakhirkan paket untuk menerima versi terbaru.

```
$ sudo apt-get upgrade -y
```

4. Boot ulang untuk memuat versi kernel terbaru.

```
$ sudo reboot
```

5. Hubungkan kembali ke instans Anda setelah boot ulang.
6. Instal paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
$ sudo apt-get install -y linux-headers-$(uname -r) linux-modules-extra-$(uname -r)
```

7. Nonaktifkan driver nouveau sumber terbuka untuk kartu grafis NVIDIA.

   a. Menambahkan nouveau ke file daftar hitam /etc/modprobe.d/blacklist.conf. Salin blok kode berikut dan tempelkan ke terminal.

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

   b. Edit file /etc/default/grub dan tambahkan baris berikut:

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

   c. Bangun kembali konfigurasi Grub.

```
$ sudo update-grub
```

8. Unduh utilitas instalasi driver GRID menggunakan perintah berikut:

```
$ aws s3 cp --recursive s3://ec2-linux-nvidia-drivers/latest/ .
```

   Beberapa versi driver GRID disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut.

```
$ aws s3 ls --recursive s3://ec2-linux-nvidia-drivers/
```

9. Tambahkan izin untuk menjalankan utilitas penginstalan driver menggunakan perintah berikut.

```
$ chmod +x NVIDIA-Linux-x86_64*.run
```

10. Jalankan skrip instalasi mandiri sebagai berikut untuk menginstal driver GRID yang Anda unduh. Misalnya:

    ```
    $ sudo /bin/sh ./NVIDIA-Linux-x86_64*.run
    ```

    Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

11. Konfirmasikan bahwa pengemudi berfungsi. Respons untuk perintah berikut mencantumkan versi driver NVIDIA yang diinstal dan detail tentang GPU.

    ```
    $ nvidia-smi -q | head
    ```

12. Jika Anda menggunakan perangkat lunak NVIDIA vGPU versi 14.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

    ```
    $ sudo touch /etc/modprobe.d/nvidia.conf
    ```

    ```
    $ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/
    modprobe.d/nvidia.conf
    ```

13. Boot ulang instans.

    ```
    $ sudo reboot
    ```

14. (Opsional) Bergantung pada kasus penggunaan Anda, Anda dapat menyelesaikan langkah opsional berikut. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah-langkah ini.

    a. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

    b. Mode NVIDIA Quadro Virtual Workstation diaktifkan secara default. Untuk mengaktifkan Aplikasi Virtual GRID untuk kemampuan hosting Aplikasi RDSH, selesaikan langkah-langkah aktivasi Aplikasi Virtual GRID di Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda.

c. Instal desktop/workstation paket GUI.

```
$ sudo apt-get install -y lightdm ubuntu-desktop
```

Sistem operasi Windows

Untuk menginstal driver NVIDIA GRID pada instans Windows Anda

1. Hubungkan ke instance Windows Anda dan buka PowerShell jendela.
2. Unduh driver dan Perjanjian Lisensi Pengguna Akhir NVIDIA GRID Cloud dari Amazon S3 ke desktop Anda menggunakan PowerShell perintah berikut.

```
$Bucket = "ec2-windows-nvidia-drivers"
$KeyPrefix = "latest"
$LocalPath = "$home\Desktop\NVIDIA"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
$LocalFileName = $Object.Key
if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
$LocalFilePath = Join-Path $LocalPath $LocalFileName
Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
}
}
```

   Banyak versi driver NVIDIA GRID disimpan dalam bucket ini. Anda dapat mengunduh semua versi Windows yang tersedia di bucket dengan menghapus opsi `-KeyPrefix $KeyPrefix`. Untuk informasi tentang versi driver NVIDIA GRID untuk sistem operasi Anda, lihat Perangkat Lunak NVIDIA Virtual GPU (vGPU) di situs web NVIDIA.

   Dimulai dengan GRID versi 11.0, Anda dapat menggunakan driver di `latest` untuk instans G3 dan G4dn. Kami tidak akan menambahkan versi setelah 11.0 hingga `g4/latest`, tetapi akan mempertahankan versi 11.0 dan versi sebelumnya khusus untuk G4dn di `g4/latest`.

   Instans G5 memerlukan GRID 13.1 atau setelahnya (atau GRID 12.4 atau setelahnya).
3. Arahkan ke desktop dan buka file instalasi untuk meluncurkannya (pilih versi driver yang sesuai dengan versi OS instans Anda). Ikuti petunjuk untuk menginstal driver dan mem-boot ulang

instans Anda sesuai kebutuhan. Untuk memverifikasi bahwa GPU berfungsi dengan benar, periksa Device Manager.

4. (Opsional) Gunakan perintah berikut untuk menonaktifkan halaman lisensi di panel kontrol untuk mencegah pengguna mengubah jenis produk secara tidak sengaja (NVIDIA GRID Virtual Workstation diaktifkan secara default). Untuk informasi selengkapnya, lihat Panduan Pengguna Lisensi GRID.

   PowerShell

   Jalankan PowerShell perintah berikut untuk membuat nilai registri untuk menonaktifkan halaman lisensi di panel kontrol. AMI Alat AWS untuk PowerShell di AWS Windows default ke versi 32-bit dan perintah ini gagal. Sebagai gantinya, gunakan versi 64-bit dari PowerShell disertakan dengan sistem operasi.

   ```
   New-Item -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global" -Name GridLicensing
   New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" -
   Name "NvCplDisableManageLicensePage" -PropertyType "DWord" -Value "1"
   ```

   Command Prompt

   Jalankan perintah registri berikut untuk membuat nilai registri untuk menonaktifkan halaman lisensi di panel kontrol. Anda dapat menjalankannya menggunakan jendela Command Prompt atau versi 64-bitPowerShell.

   ```
   reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global\GridLicensing" /v
    NvCplDisableManageLicensePage /t REG_DWORD /d 1
   ```

5. (Opsional) Bergantung pada kasus penggunaan Anda, Anda dapat menyelesaikan langkah opsional berikut. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah-langkah ini.

   a. Untuk membantu memanfaatkan empat tampilan resolusi hingga 4K, siapkan protokol tampilan berkinerja tinggi, Amazon DCV.

   b. Mode NVIDIA Quadro Virtual Workstation diaktifkan secara default. Untuk mengaktifkan Aplikasi Virtual GRID untuk kemampuan hosting Aplikasi RDSH, selesaikan langkah-langkah aktivasi Aplikasi Virtual GRID di Aktifkan Aplikasi Virtual NVIDIA GRID di instans Amazon EC2 GPU-based Anda.

## Instal driver gaming NVIDIA (instans G7e, G6, G6e, G5, dan G4dn)

Driver ini hanya tersedia untuk AWS pelanggan. Dengan mengunduhnya, Anda setuju untuk menggunakan perangkat lunak yang diunduh hanya untuk mengembangkan AMI untuk digunakan dengan perangkat keras RTX PRO 6000 Blackwell, NVIDIA L4, NVIDIA L40S, NVIDIA A10G, NVIDIA Tesla T4, atau NVIDIA Tesla M60. Anda dapat menggunakan driver GRID untuk membuat dan menggunakan AMI dalam AWS lingkungan. Setelah menginstal perangkat lunak, Anda terikat oleh persyaratan Perjanjian Lisensi Pengguna Akhir Cloud NVIDIA GRID.

### Pertimbangan-pertimbangan

- Instans G3 memerlukan resolusi DNS yang AWS disediakan agar lisensi GRID berfungsi.
- IMDSv2 hanya didukung dengan driver NVIDIA versi 495.x atau lebih tinggi.

### Prasyarat

- (Linux) Verifikasi bahwa AWS CLI diinstal pada instans Anda dan dikonfigurasi dengan kredenSIAL default. Untuk informasi selengkapnya, lihat Menginstal AWS CLI dalam Panduan Pengguna AWS Command Line Interface .
- Pengguna atau peran Anda harus memiliki izin yang diberikan yang berisi AmazonS3ReadOnlyAccess kebijakan.

### Amazon Linux 2023

Untuk menginstal driver game NVIDIA pada instans Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

   ```
   [ec2-user ~]$ sudo dnf update -y
   ```

2. Install gcc dan make, jika belum terinstal.

   ```
   [ec2-user ~]$ sudo dnf install gcc make
   ```

3. Boot ulang untuk memuat versi kernel terbaru.

   ```
   [ec2-user ~]$ sudo reboot
   ```

4. Sambungkan kembali ke instance Anda setelah di-boot ulang.
5. Instal paket header kernel.

```
[ec2-user ~]$ sudo dnf install -y kernel-devel kernel-modules-extra kernel-devel-
$(uname -r) kernel-headers-$(uname -r) dkms
```

6. Unduh utilitas instalasi driver game menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

   Beberapa versi driver game disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

7. Ekstrak utilitas instalasi driver game dari `.zip` arsip yang diunduh.

```
[ec2-user ~]$ unzip latest-driver-name.zip -d nvidia-drivers
```

8. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

9. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo ./nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

   Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

10. Gunakan perintah berikut untuk membuat file konfigurasi yang diperlukan.

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

11. Gunakan perintah berikut untuk mengunduh dan mengganti nama file sertifikasi.
    - Untuk versi 590.48 atau yang lebih baru:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert_2026_03_02.cert"
```

- Untuk versi 575.64 hingga 580.105.08:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2025_07_01.cert"
```

- Untuk versi 460.39 hingga 575.57:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2024_02_22.cert"
```

- Untuk versi 440.68 hingga 445.48:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- Untuk versi sebelumnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

12. Jika Anda menggunakan driver NVIDIA versi 510.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

13. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

14. Verifikasi lisensi NVIDIA Gaming menggunakan perintah berikut.

```
[ec2-user ~]$ nvidia-smi.exe -q
```

Dalam output, carivGPU Software Licensed Product.

15. (Opsional) Untuk membantu memanfaatkan satu tampilan hingga resolusi 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

Amazon Linux 2

Untuk menginstal driver game NVIDIA pada instans Anda

1. Terhubung ke instans Anda. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

2. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install gcc make
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Sambungkan kembali ke instance Anda setelah di-boot ulang.
5. Instal paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y kernel-devel-$(uname -r)
```

6. Unduh utilitas instalasi driver game menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

Beberapa versi driver game disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

7. Ekstrak utilitas instalasi driver game dari `.zip` arsip yang diunduh.

```
[ec2-user ~]$ unzip latest-driver-name.zip -d nvidia-drivers
```

8. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

9. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo ./nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

   Jika Anda menggunakan Amazon Linux 2 dengan kernel versi 5.10, gunakan perintah berikut untuk menginstal driver gaming NVIDIA.

```
[ec2-user ~]$ sudo CC=/usr/bin/gcc10-cc ./NVIDIA-Linux-x86_64*.run
```

   Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

10. Gunakan perintah berikut untuk membuat file konfigurasi yang diperlukan.

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

11. Gunakan perintah berikut untuk mengunduh dan mengganti nama file sertifikasi.
    - Untuk versi 590.48 atau yang lebih baru:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert_2026_03_02.cert"
```

    - Untuk versi 575.64 hingga 580.105.08:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2025_07_01.cert"
```

    - Untuk versi 460.39 hingga 575.57:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2024_02_22.cert"
```

    - Untuk versi 440.68 hingga 445.48:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- Untuk versi sebelumnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

12. Jika Anda menggunakan driver NVIDIA versi 510.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

13. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

14. Verifikasi lisensi NVIDIA Gaming menggunakan perintah berikut.

```
[ec2-user ~]$ nvidia-smi.exe -q
```

Dalam output, carivGPU `Software Licensed Product`.

15. (Opsional) Untuk membantu memanfaatkan satu tampilan hingga resolusi 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

CentOS 7 dan Red Hat Enterprise Linux 7

Untuk menginstal driver game NVIDIA pada instans Anda

1. Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y unzip kernel-devel-$(uname -r)
```

6. Nonaktifkan driver nouveau sumber terbuka untuk kartu grafis NVIDIA.

   a. Menambahkan nouveau ke file daftar hitam /etc/modprobe.d/blacklist.conf. Salin blok kode berikut dan tempelkan ke terminal.

   ```
   [ec2-user ~]$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
   blacklist vga16fb
   blacklist nouveau
   blacklist rivafb
   blacklist nvidiafb
   blacklist rivatv
   EOF
   ```

   b. Edit file /etc/default/grub dan tambahkan baris berikut:

   ```
   GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
   ```

   c. Bangun kembali konfigurasi Grub.

   ```
   [ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

7. Unduh utilitas instalasi driver game menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

Beberapa versi driver game disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

8. Ekstrak utilitas instalasi driver game dari `.zip` arsip yang diunduh.

```
[ec2-user ~]$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

9. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

10. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

    Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

11. Gunakan perintah berikut untuk membuat file konfigurasi yang diperlukan.

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

12. Gunakan perintah berikut untuk mengunduh dan mengganti nama file sertifikasi.
    - Untuk versi 590.48 atau yang lebih baru:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert_2026_03_02.cert"
```

    - Untuk versi 575.64 hingga 580.105.08:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2025_07_01.cert"
```

    - Untuk versi 460.39 hingga 575.57:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2024_02_22.cert"
```

    - Untuk versi 440.68 hingga 445.48:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- Untuk versi sebelumnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

13. Jika Anda menggunakan driver NVIDIA versi 510.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

14. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

15. (Opsional) Untuk membantu memanfaatkan satu tampilan hingga resolusi 4K, atur protokol tampilan kinerja tinggi Amazon DCV. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah ini.

CentOS Stream 8 dan Red Hat Enterprise Linux 8

Untuk menginstal driver game NVIDIA pada instans Anda

1. Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo yum install -y unzip kernel-devel-$(uname -r)
```

6. Unduh utilitas instalasi driver game menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

   Beberapa versi driver game disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

7. Ekstrak utilitas instalasi driver game dari `.zip` arsip yang diunduh.

```
[ec2-user ~]$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

8. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

9. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

   Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

10. Gunakan perintah berikut untuk membuat file konfigurasi yang diperlukan.

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

11. Gunakan perintah berikut untuk mengunduh dan mengganti nama file sertifikasi.
    - Untuk versi 590.48 atau yang lebih baru:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert_2026_03_02.cert"
```

- Untuk versi 575.64 hingga 580.105.08:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2025_07_01.cert"
```

- Untuk versi 460.39 hingga 575.57:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2024_02_22.cert"
```

- Untuk versi 440.68 hingga 445.48:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
```

- Untuk versi sebelumnya:

```
[ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
```

12. Jika Anda menggunakan driver NVIDIA versi 510.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

13. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

14. (Opsional) Untuk membantu memanfaatkan satu tampilan hingga resolusi 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

Rocky Linux 8

Untuk menginstal driver game NVIDIA pada instans Anda

1. Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.

```
[ec2-user ~]$ sudo yum install -y gcc make
```

2. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
[ec2-user ~]$ sudo yum update -y
```

3. Boot ulang untuk memuat versi kernel terbaru.

```
[ec2-user ~]$ sudo reboot
```

4. Hubungkan kembali ke instans Anda setelah boot ulang.
5. Instal paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
[ec2-user ~]$ sudo dnf install -y unzip elfutils-libelf-devel libglvnd-devel
 kernel-devel-$(uname -r)
```

6. Unduh utilitas instalasi driver game menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

   Beberapa versi driver game disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut:

```
[ec2-user ~]$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

7. Ekstrak utilitas instalasi driver game dari `.zip` arsip yang diunduh.

```
[ec2-user ~]$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

8. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
[ec2-user ~]$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

9. Jalankan penginstal menggunakan perintah berikut:

```
[ec2-user ~]$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

10. Gunakan perintah berikut untuk membuat file konfigurasi yang diperlukan.

```
[ec2-user ~]$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

11. Gunakan perintah berikut untuk mengunduh dan mengganti nama file sertifikasi.
    - Untuk versi 590.48 atau yang lebih baru:

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
      gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert_2026_03_02.cert"
      ```

    - Untuk versi 575.64 hingga 580.105.08:

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
      gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2025_07_01.cert"
      ```

    - Untuk versi 460.39 hingga 575.57:

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
      gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2024_02_22.cert"
      ```

    - Untuk versi 440.68 hingga 445.48:

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
      gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
      ```

    - Untuk versi sebelumnya:

      ```
      [ec2-user ~]$ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
      gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
      ```

12. Jika Anda menggunakan driver NVIDIA versi 510.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
[ec2-user ~]$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
[ec2-user ~]$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /
etc/modprobe.d/nvidia.conf
```

13. Boot ulang instans.

```
[ec2-user ~]$ sudo reboot
```

14. (Opsional) Untuk membantu memanfaatkan satu tampilan hingga resolusi 4K, atur protokol tampilan kinerja tinggi Amazon DCV.

Ubuntu dan Debian

Untuk menginstal driver game NVIDIA pada instans Anda

1. Hubungkan dengan instans Linux Anda. Install gcc dan make, jika belum terinstal.

```
$ sudo apt-get install -y gcc make build-essential
```

2. Perbarui cache paket Anda dan dapatkan pembaruan paket untuk instans Anda.

```
$ sudo apt-get update -y
```

3. Mutakhirkan paket `linux-aws` untuk menerima versi terbaru.

```
$ sudo apt-get upgrade -y linux-aws
```

4. Boot ulang untuk memuat versi kernel terbaru.

```
$ sudo reboot
```

5. Hubungkan kembali ke instans Anda setelah boot ulang.
6. Instal paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
$ sudo apt install -y unzip dkms linux-headers-$(uname -r)
```

7. Nonaktifkan driver `nouveau` sumber terbuka untuk kartu grafis NVIDIA.

a. Menambahkan nouveau ke file daftar hitam `/etc/modprobe.d/blacklist.conf`. Salin blok kode berikut dan tempelkan ke terminal.

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

b. Edit file `/etc/default/grub` dan tambahkan baris berikut:

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

c. Bangun kembali konfigurasi Grub.

```
$ sudo update-grub
```

8. Unduh utilitas instalasi driver game menggunakan perintah berikut:

```
$ aws s3 cp --recursive s3://nvidia-gaming/linux/latest/ .
```

Beberapa versi driver game disimpan dalam bucket ini. Anda dapat melihat semua versi yang tersedia menggunakan perintah berikut:

```
$ aws s3 ls --recursive s3://nvidia-gaming/linux/
```

9. Ekstrak utilitas instalasi driver game dari `.zip` arsip yang diunduh.

```
$ unzip *Gaming-Linux-Guest-Drivers.zip -d nvidia-drivers
```

10. Tambahkan izin untuk menjalankan utilitas instalasi driver menggunakan perintah berikut:

```
$ chmod +x nvidia-drivers/NVIDIA-Linux-x86_64*-grid.run
```

11. Jalankan penginstal menggunakan perintah berikut:

```
$ sudo nvidia-drivers/NVIDIA-Linux-x86_64*.run
```

Saat diminta, terima perjanjian lisensi dan tentukan opsi penginstalan yang diperlukan (Anda dapat menerima opsi default).

12. Gunakan perintah berikut untuk membuat file konfigurasi yang diperlukan.

```
$ cat << EOF | sudo tee -a /etc/nvidia/gridd.conf
vGamingMarketplace=2
EOF
```

13. Gunakan perintah berikut untuk mengunduh dan mengganti nama file sertifikasi.

    - Untuk versi 590.48 atau yang lebih baru:

    ```
    $ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
    gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert_2026_03_02.cert"
    ```

    - Untuk versi 575.64 hingga 580.105.08:

    ```
    $ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
    gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2025_07_01.cert"
    ```

    - Untuk versi 460.39 hingga 575.57:

    ```
    $ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
    gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCertLinux_2024_02_22.cert"
    ```

    - Untuk versi 440.68 hingga 445.48:

    ```
    $ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
    gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2020_04.cert"
    ```

    - Untuk versi sebelumnya:

    ```
    $ sudo curl -o /etc/nvidia/GridSwCert.txt "https://nvidia-
    gaming.s3.amazonaws.com/GridSwCert-Archive/GridSwCert-Linux_2019_09.cert"
    ```

14. Jika Anda menggunakan driver NVIDIA versi 510.x atau lebih tinggi pada instans G4dn, G5, atau G5G, nonaktifkan GSP dengan perintah berikut. Untuk informasi lebih lanjut tentang mengapa ini diperlukan, lihat dokumentasi NVIDIA.

```
$ sudo touch /etc/modprobe.d/nvidia.conf
```

```
$ echo "options nvidia NVreg_EnableGpuFirmware=0" | sudo tee --append /etc/
modprobe.d/nvidia.conf
```

15. Boot ulang instans.

```
$ sudo reboot
```

16. (Opsional) Untuk membantu memanfaatkan satu tampilan hingga resolusi 4K, atur protokol tampilan kinerja tinggi Amazon DCV. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah ini.

Sistem operasi Windows

Sebelum Anda menginstal driver gaming NVIDIA pada instance Anda, Anda harus memastikan bahwa prasyarat berikut terpenuhi selain pertimbangan yang disebutkan untuk semua driver game.

- Jika Anda meluncurkan instance Windows menggunakan AMI Windows khusus, AMI harus berupa gambar standar yang dibuat dengan Windows Sysprep untuk memastikan bahwa driver game berfungsi. Untuk informasi selengkapnya, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.
- Konfigurasikan kredenSIAL default untuk AWS Tools for Windows PowerShell pada instans Windows Anda. Untuk informasi selengkapnya, lihat Memulai AWS Tools for Windows PowerShell di Panduan Pengguna.Alat AWS untuk PowerShell

Untuk menginstal driver game NVIDIA pada instans Windows Anda

1. Hubungkan ke instance Windows Anda dan buka PowerShell jendela.
2. Unduh dan instal driver game menggunakan perintah PowerShell berikut ini.

```
$Bucket = "nvidia-gaming"
$KeyPrefix = "windows/latest"
$LocalPath = "$home\Desktop\NVIDIA"
$Objects = Get-S3Object -BucketName $Bucket -KeyPrefix $KeyPrefix -Region us-east-1
foreach ($Object in $Objects) {
$LocalFileName = $Object.Key
if ($LocalFileName -ne '' -and $Object.Size -ne 0) {
$LocalFilePath = Join-Path $LocalPath $LocalFileName
```

```
Copy-S3Object -BucketName $Bucket -Key $Object.Key -LocalFile $LocalFilePath -
Region us-east-1
}
}
```

Beberapa versi driver NVIDIA GRID disimpan dalam bucket S3 ini. Anda dapat mengunduh semua versi yang tersedia di bucket jika Anda mengubah nilai `$KeyPrefix` variabel dari "windows/latest" ke “windows”.

3. Arahkan ke desktop dan buka file instalasi untuk meluncurkannya (pilih versi driver yang sesuai dengan versi OS instans Anda). Ikuti petunjuk untuk menginstal driver dan mem-boot ulang instans Anda sesuai kebutuhan. Untuk memverifikasi bahwa GPU berfungsi dengan benar, periksa Device Manager.
4. Gunakan salah satu metode berikut untuk mendaftarkan driver.

   Version 527.27 or above

   Buat kunci registri berikut dengan versi 64-bit PowerShell, atau jendela Command Prompt.

   kunci: `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global`

   nama: v GamingMarketplace

   tipe: DWord

   nilai: 2

   PowerShell

   Jalankan PowerShell perintah berikut untuk membuat nilai registri ini. AMI Alat AWS untuk PowerShell di AWS Windows default ke versi 32-bit dan perintah ini gagal. Sebagai gantinya, gunakan versi 64-bit yang PowerShell disertakan dengan sistem operasi.

   ```
   New-ItemProperty -Path "HKLM:\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global"
    -Name "vGamingMarketplace" -PropertyType "DWord" -Value "2"
   ```

   Command Prompt

   Jalankan perintah registri berikut untuk membuat nilai registri ini. Anda dapat menjalankannya menggunakan jendela Command Prompt atau versi 64-bit PowerShell.

```
reg add "HKLM\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global" /v
 vGamingMarketplace /t REG_DWORD /d 2
```

Earlier versions

Buat kunci registri berikut dengan versi 64-bit PowerShell, atau jendela Command Prompt.

kunci: `HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global`

nama: v GamingMarketplace

tipe: DWord

nilai: 2

PowerShell

Jalankan PowerShell perintah berikut untuk membuat nilai registri ini. AMI Alat AWS untuk PowerShell di AWS Windows default ke versi 32-bit dan perintah ini gagal. Sebagai gantinya, gunakan versi 64-bit yang PowerShell disertakan dengan sistem operasi.

```
New-ItemProperty -Path "HKLM:\SOFTWARE\NVIDIA Corporation\Global" -Name
 "vGamingMarketplace" -PropertyType "DWord" -Value "2"
```

Command Prompt

Jalankan perintah registri berikut untuk membuat kunci registri ini dengan jendela Command Prompt. Anda juga dapat menggunakan perintah ini dalam versi 64-bitPowerShell.

```
reg add "HKLM\SOFTWARE\NVIDIA Corporation\Global" /v vGamingMarketplace /t
 REG_DWORD /d 2
```

5. Jalankan perintah berikut di PowerShell. Hal ini akan mengunduh file sertifikasi, mengganti nama file `GridSwCert.txt`, dan memindahkan file ke folder Dokumen Publik di drive sistem Anda. Biasanya, jalur foldernya adalah `C:\Users\Public\Documents`.
   - Untuk versi 591.59 atau yang lebih baru:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-
Archive/GridSwCert_2026_03_02.cert" -OutFile "$Env:PUBLIC\Documents
\GridSwCert.txt"
```

- Untuk versi 576.80 hingga 581.80:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-
Archive/GridSwCertWindows_2025_07_01.cert" -OutFile "$Env:PUBLIC\Documents
\GridSwCert.txt"
```

- Untuk versi 460.39 hingga 576.72:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-
Archive/GridSwCertWindows_2024_02_22.cert" -OutFile "$Env:PUBLIC\Documents
\GridSwCert.txt"
```

- Untuk versi 445.87:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-
Archive/GridSwCert-Windows_2020_04.cert" -OutFile "$Env:PUBLIC\Documents
\GridSwCert.txt"
```

- Untuk versi sebelumnya:

```
Invoke-WebRequest -Uri "https://nvidia-gaming.s3.amazonaws.com/GridSwCert-
Archive/GridSwCert-Windows_2019_09.cert" -OutFile "$Env:PUBLIC\Documents
\GridSwCert.txt"
```

Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
```

6. Booting ulang instans Anda.
7. Cari `nvidia-smi.exe` file pada instance.

```
Get-ChildItem -Path C:\ -Recurse -Filter "nvidia-smi.exe"
```

Verifikasi lisensi NVIDIA Gaming menggunakan perintah berikut. Ganti *path* dengan nama folder dalam output dari perintah sebelumnya.

```
C:\Windows\System32\DriverStore\FileRepository\path\nvidia-smi.exe -q
```

Output harus serupa dengan yang berikut ini.

```
vGPU Software Licensed Product
Product Name                : NVIDIA Cloud Gaming
License Status              : Licensed (Expiry: N/A)
```

8. (Opsional) Untuk membantu memanfaatkan tampilan tunggal hingga resolusi 4K, atur protokol tampilan kinerja tinggi Amazon DCV. Jika Anda tidak memerlukan fungsionalitas ini, jangan selesaikan langkah ini.

## Instal driver ENA pada instans EC2 Windows

Jika instans Anda tidak didasarkan pada salah satu Windows Amazon Machine Images (AMI) terbaru yang disediakan Amazon, gunakan prosedur berikut untuk menginstal driver ENA saat ini pada instans Anda. Anda harus melakukan pembaruan ini pada saat yang tepat untuk mem-boot ulang instans Anda. Jika skrip penginstalan tidak secara otomatis me-reboot instans Anda, kami sarankan Anda me-reboot instans sebagai langkah terakhir.

Jika Anda menggunakan volume penyimpanan instans untuk menyimpan data saat instans berjalan, data tersebut akan dihapus saat Anda menghentikan instans. Sebelum Anda menghentikan instans Anda, verifikasi bahwa Anda telah menyalin data apa pun yang Anda perlukan dari volume penyimpanan instans Anda ke penyimpanan persisten, seperti Amazon EBS atau Amazon S3.

### Prasyarat

Untuk menginstal atau meng-upgrade driver ENA, instans Windows Anda harus memenuhi prasyarat berikut ini:

- PowerShell versi 3.0 atau yang lebih baru diinstal.
- Perintah yang ditunjukkan pada bagian ini harus berjalan dalam versi 64-bitPowerShell. Jangan gunakan x86 versi PowerShell. Itu adalah versi 32-bit dari shell, dan tidak didukung untuk perintah ini.

## Langkah 1: Mencadangkan data Anda

Kami menyarankan Anda membuat AMI cadangan, jika Anda tidak dapat mengembalikan perubahan Anda melalui Device Manager. Untuk membuat AMI cadangan dengan Konsol Manajemen AWS, ikuti langkah-langkah berikut:

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans yang memerlukan peningkatan driver, dan pilih Hentikan instans dari menu Status instans.
4. Setelah instans dihentikan, pilih instans lagi. Untuk membuat cadangan, pilih Gambar dan templat dari menu Tindakan, lalu pilih Buat gambar.
5. Untuk memulai ulang instans Anda, pilih Mulai instans dari menu status Instans.

## Langkah 2: Menginstal atau meningkatkan driver ENA Anda

Anda dapat menginstal atau meningkatkan driver ENA Anda dengan AWS Systems Manager Distributor, atau dengan PowerShell cmdlet. Untuk petunjuk selengkapnya, pilih tab yang cocok dengan metode yang ingin Anda gunakan.

Systems Manager Distributor

Anda dapat menggunakan fitur Distributor Systems Manager untuk menyebarkan paket ke simpul terkelola Systems Manager Anda. Dengan Systems Manager Distributor, Anda dapat menginstal paket driver ENA satu kali, atau dengan pembaruan terjadwal. Untuk informasi selengkapnya tentang cara menginstal paket driver ENA (`AwsEnaNetworkDriver`) dengan Systems Manager Distributor, lihat Menginstal atau memperbarui paket di Panduan Pengguna AWS Systems Manager .

PowerShell

Bagian ini mencakup cara mengunduh dan menginstal paket driver ENA pada instans Anda dengan PowerShell cmdlet.

Opsi 1: Mengunduh dan mengekstraksi versi terbaru

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
2. Gunakan cmdlet invoke-webrequest untuk mengunduh paket driver terbaru:

```
PS C:\> invoke-webrequest https://ec2-windows-drivers-
downloads.s3.amazonaws.com/ENA/Latest/AwsEnaNetworkDriver.zip -
outfile $env:USERPROFILE\AwsEnaNetworkDriver.zip
```

> **Note**
>
> Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:
>
> ```
> [Net.ServicePointManager]::SecurityProtocol =
>  [Net.SecurityProtocolType]::Tls12
> ```
>
> Atau, Anda dapat mengunduh paket driver terbaru dari jendela browser di instans Anda.

3. Gunakan cmdlet expand-archive untuk mengekstrak arsip zip yang Anda unduh ke instans Anda:

```
PS C:\> expand-archive $env:userprofile\AwsEnaNetworkDriver.zip -
DestinationPath $env:userprofile\AwsEnaNetworkDriver
```

Opsi 2: Mengunduh dan mengekstraksi versi tertentu

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
2. Unduh paket driver ENA untuk versi tertentu yang Anda inginkan dari tautan versi di tabel RIWAYAT VERSI DRIVER ENA Windows.
3. Ekstrak arsip zip ke instans Anda.

Instal driver ENA dengan PowerShell

Langkah-langkah penginstalan sama apakah Anda telah mengunduh driver terbaru atau versi tertentu. Untuk menginstal driver ENA, ikuti langkah-langkah ini.

1. Untuk menginstal driver, jalankan `install.ps1` PowerShell skrip dari `AwsEnaNetworkDriver` direktori pada instance Anda. Jika Anda mendapatkan kesalahan, pastikan Anda menggunakan PowerShell 3.0 atau yang lebih baru.
2. Jika penginstal tidak secara otomatis me-reboot instance Anda, jalankan Restart-Computer PowerShell cmdlet.

```
PS C:\> Restart-Computer
```

### Langkah 3 (opsional): Verifikasi versi driver ENA setelah instalasi

Untuk memastikan bahwa paket driver ENA berhasil diinstal pada instans Anda, Anda dapat memverifikasi versi baru sebagai berikut:

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
2. Untuk membuka Windows Device Manager, masukkan `devmgmt.msc` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Device Manager.
4. Pilih panah di sebelah kiri Adaptor jaringan untuk memperluas daftar.
5. Pilih nama, atau buka menu konteks untuk Adaptor Jaringan Elastis Amazon, lalu pilih Properti. Ini membuka dialog Properti Adaptor Jaringan Elastis Amazon.

> **Note**
>
> Adaptor ENA semua menggunakan driver yang sama. Jika Anda memiliki beberapa adaptor ENA, Anda dapat memilih salah satu dari mereka untuk memperbarui driver untuk semua adaptor ENA.

6. Untuk memverifikasi versi saat ini yang diinstal, buka tab Driver dan periksa Versi Driver. Jika versi saat ini tidak cocok dengan versi target Anda, lihat Memecahkan masalah driver Windows Adaptor Jaringan Elastic.

## Putar kembali instalasi driver ENA

Jika ada yang salah dengan instalasi, Anda mungkin perlu memutar kembali driver. Ikuti langkah-langkah ini untuk memutar kembali ke versi driver ENA sebelumnya yang diinstal pada instans Anda.

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.

2. Untuk membuka Windows Device Manager, masukkan `devmgmt.msc` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Device Manager.
4. Pilih panah di sebelah kiri Adaptor jaringan untuk memperluas daftar.
5. Pilih nama, atau buka menu konteks untuk Adaptor Jaringan Elastis Amazon, lalu pilih Properti. Ini membuka dialog Properti Adaptor Jaringan Elastis Amazon.

> **Note**
>
> Adaptor ENA semua menggunakan driver yang sama. Jika Anda memiliki beberapa adaptor ENA, Anda dapat memilih salah satu dari mereka untuk memperbarui driver untuk semua adaptor ENA.

6. Untuk memutar kembali driver, buka tab Driver dan pilih Roll Back Driver. Ini membuka jendela rollback Driver Package.

> **Note**
>
> Jika tab Driver tidak menampilkan tindakan Roll Back Driver, atau jika tindakan tidak tersedia, itu berarti bahwa Driver Store pada instans Anda tidak berisi paket driver yang diinstal sebelumnya. Untuk memecahkan masalah ini, lihat Skenario pemecahan masalah, dan perluas bagian yang diinstal versi driver ENA yang tidak terduga. Untuk informasi selengkapnya tentang proses pemilihan paket driver perangkat, lihat Cara Windows memilih paket driver untuk perangkat di situs web dokumentasi Microsoft.

## Lacak rilis versi driver ENA Windows

AMI Windows menyertakan driver Windows ENA untuk mengaktifkan jaringan yang ditingkatkan.

Untuk Windows Server versi 2016 dan yang lebih tinggi, kami sarankan Anda menggunakan versi driver terbaru. Untuk versi Windows Server sebelumnya, lihat tabel berikut untuk menentukan versi driver ENA mana yang akan digunakan.

| Versi Windows Server | Versi driver ENA |
|---|---|
| Windows Server 2012 R2 | 2.6.0 dan sebelumnya |
| Windows Server 2012 | 2.6.0 dan sebelumnya |

| Versi Windows Server | Versi driver ENA |
| --- | --- |
| Windows Server 2008 R2 | 2.2.3 dan sebelumnya |

RIWAYAT VERSI DRIVER ENA Windows

Tabel berikut merangkum perubahan untuk setiap rilis.

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.11.0 | Fitur baru<br>• Menambahkan dukungan untuk alokasi antrian dinamis dengan penskalaan hingga I/O 64 antrian, tergantung pada batas jenis instans.<br><br>Perbaikan bug<br>• Memperbaiki penanganan kondisi kesalahan yang hilang saat mengambil lebar alamat DMA fisik dari perangkat. | Agustus 1, 2025 |
| 2.0.0 | Perbaikan bug<br>• Memperbaiki penanganan kegagalan alokasi memori yang tidak tepat selama inisialisasi antrian untuk mencegah reboot yang tidak terduga.<br>• Memperbaiki kemungkinan kesalahan konfigura si offload jaringan di mana kemampuan perangkat diabaikan. | Juni 24, 2025 |
| 2.9.0 | | Desember 12, 2024 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| | Fitur baru<br>• Menambahkan dukungan untuk permintaan reset asinkron yang diprakarsai oleh perangkat.<br>• Menambahkan dukungan untuk menangani nilai kedalaman LLQ besar maksimum yang disediakan oleh perangkat.<br>• Menambahkan ID Per `58001` istiwa di Windows Event Viewer untuk meningkatkan visibilitas ke transisi status daya tak terduga yang disebabkan oleh kesalahan konfigurasi perangkat.<br><br>Perbaikan bug<br>• Memperbaiki penanganan kegagalan alokasi memori yang tidak tepat selama inisialisasi perangkat untuk mencegah reboot yang tidak terduga.<br>• Memperbaiki masalah dalam rutinitas layanan interupsi yang dapat mengantri DPC selama perangkat berhenti, mencegah reboot yang tidak terduga. | |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.8.0 | Perbaikan bug<br>• Memperbaiki kondisi balapan dalam aliran lengkap pemrosesan daftar buffer jaringan keluar (NBL), yang dapat menyebabkan kerusakan memori yang disebabkan oleh upaya merilis NBL yang sudah dirilis.<br>• Memperbaiki kesalahan deteksi protokol L3 saat menonaktifkan semua offload LSO dan checksum yang dapat menyebabkan perilaku tak terduga. | 30. September 2024 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.7.0 | Fitur baru<br>• Menghapus dukungan untuk Windows Server 2012 (Windows 8) dan Windows Server 2012 R2 (Windows 8.1). Versi sistem operasi ini telah mencapai akhir dukungan dari AWS. Penginstalan driver akan gagal pada Windows Server 2012 dan sebelumnya.<br>• Menambahkan dukungan untuk menurunkan perhitungan checksum IPv6 Tx ke perangkat.<br>• Menambahkan dukungan Low Latency Queuing (LLQ) yang luas. Ini diaktifkan secara dinamis berdasarkan rekomendasi perangkat. Anda dapat mengganti pengaturan ini dengan kunci registri “WideLLQ” baru.<br>• Menambahkan pelaporan untuk drop paket akibat overrun Rx, yang menunjukkan ruang yang tidak mencukupi di ring Rx untuk paket yang masuk.<br>• Menambahkan dukungan untuk pemberitahuan konfigurasi suboptimal dari perangkat. Lihat ID peristiwa `59000` di penampil acara Windows.<br><br>Perbaikan bug<br>• Hindari reset perangkat yang tidak perlu yang disebabkan oleh paket Tx dengan header yang melebihi ukuran header Low Latency Queuing (LLQ) maksimum. | 1 Mei 2024 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.6.0 | Fitur baru<br>• Menambahkan metrik performa jaringan berikut untuk tipe instans yang mendukung ENA Ekspres.<br>  • `ena_srd_mode`<br>  • `ena_srd_tx_pkts`<br>  • `ena_srd_eligible_tx_pkts`<br>  • `ena_srd_rx_pkts`<br>  • `ena_srd_resource_utilization`<br>• Menambahkan metrik performa jaringan `conntrack_allowance_available` untuk tipe instans berbasis Nitro.<br>• Menambahkan alasan reset adaptor baru karena deteksi kerusakan data RX.<br>• Perbarui infrastruktur pencatatan driver.<br><br>Perbaikan bug<br>• Mencegah reset adaptor jika terjadi kelaparan CPU menyebabkan pembaruan metrik performa jaringan gagal.<br>• Cegah deteksi palsu interupsi pada detak jantung perangkat.<br>• | 20 Juni 2023 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| | Memperbaiki skrip instalasi driver untuk mendukung operasi downgrade.<br>• Memperbaiki statistik jumlah kesalahan penerimaan. | |
| 2.5.0 | Pengumuman<br><br>ENA Windows driver versi 2.5.0 telah dibatalkan karena kegagalan untuk menginisialisasi pada pengontrol domain Windows. Windows Client dan Windows Server tidak terpengaruh. | 17 Februari 2023 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.4.0 | Fitur baru<br>• Menambahkan dukungan untuk Windows Server 2022.<br>• Menghapus dukungan untuk Windows Server 2008 R2.<br>• Menetapkan Low Latency Queuing (LLQ) agar selalu aktif untuk meningkatkan performa pada instans Amazon EC2 generasi keenam.<br><br>Perbaikan bug<br>• Memperbaiki kegagalan untuk mempublikasikan metrik performa jaringan ke sistem Penghitung Performa untuk Windows (PCW).<br>• Memperbaiki kebocoran memori selama operasi pembacaan kunci registri.<br>• Cegah loop reset tak terbatas jika terjadi kesalahan yang tidak dapat dipulihkan selama proses reset adaptor. | 28 April 2022 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.4 | Pengumuman<br><br>ENA Windows driver versi 2.2.4 telah dibatalkan karena potensi penurunan performa pada instans EC2 generasi keenam. Kami menyarankan Anda menurunkan versi driver, menggunakan salah satu metode berikut:<br><br>• Instal versi sebelumnya<br><br>1. Unduh paket versi sebelumnya dari tautan di tabel ini (versi 2.2.3).<br>2. Jalankan skrip install.ps1 PowerShell instalasi.<br><br>Untuk detail lebih lanjut untuk langkah-langkah sebelum dan sesudah instalasi lihat Mengaktifkan jaringan yang ditingkatkan di Windows.<br><br>• Menggunakan Amazon EC2 Systems Manager untuk pembaruan massal<br><br>• Lakukan pembaruan massal melalui dokumen `AWS-ConfigureAWSPackage` SSM, dengan parameter berikut:<br>• Nama: AwsEnaNetworkDriver<br>• Versi: 2.2.3 | 26 Oktober 2021 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.3 | Fitur baru<br>• Menambahkan dukungan untuk kartu Nitro baru dengan jaringan instans 400 Gbps.<br><br>Perbaikan bug<br>• Memperbaiki kondisi balapan antara perubahan waktu sistem dan kueri waktu sistem oleh driver ENA, yang menyebabkan deteksi positif palsu dari tidak responsif nya HW.<br><br>Pengumuman<br>Driver Windows ENA versi 2.2.3 adalah versi final yang mendukung Windows Server 2008 R2. Saat ini tipe instans yang tersedia yang menggunakan ENA akan terus didukung pada Windows Server 2008 R2, dan driver tersedia dengan mengunduh. Tidak ada tipe instans masa depan yang akan mendukung Windows Server 2008 R2, dan Anda tidak dapat meluncurkan, mengimpor, atau memigrasi gambar Windows Server 2008 R2 ke tipe instans masa depan. | 25 Maret 2021 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.2 | Fitur baru<br>• Menambahkan dukungan untuk kueri metrik kinerja adaptor jaringan dengan CloudWatch dan Penghitung Kinerja untuk konsumen Windows.<br><br>Perbaikan bug<br>• Memperbaiki masalah performa pada instans bare metal. | 21 Desember 2020 |
| 2.2.1 | Fitur baru<br>• Tambahkan metode agar host bisa mengueri Elastic Network Adapter untuk metrik performa jaringan. | 1 Oktober 2020 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.0 | Fitur baru<br>• Menambahkan dukungan untuk tipe perangkat keras generasi selanjutnya.<br>• Meningkatkan waktu mulai instans setelah melanjutkan dari stop-hibernate, dan menghapus pesan kesalahan ENA positif palsu.<br><br>Optimalisasi performa<br>• Mengoptimalkan pemrosesan lalu lintas masuk.<br>• Meningkatkan manajemen memori bersama di lingkungan dengan sumber daya rendah.<br><br>Perbaikan bug<br>• Menghindari sistem crash saat penghapusan perangkat ENA dalam skenario yang jarang terjadi di mana driver gagal untuk mereset. | 12 Agustus 2020 |
| 2.1.5 | Perbaikan bug<br>• Memperbaiki kegagalan inisialisasi adaptor jaringan yang sesekali terjadi pada instans bare metal. | 23 Juni 2020 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.4 | Perbaikan bug<br>• Mencegah masalah konektivitas yang disebabkan oleh metadata paket LSO yang rusak dari tumpukan jaringan.<br>• Mencegah kerusakan sistem yang disebabkan oleh kondisi race langka yang mengakibatkan pengaksesan memori paket yang sudah dirilis. | 25 November 2019 |
| 2.1.2 | Fitur baru<br>• Menambahkan dukungan untuk laporan ID vendor untuk memungkinkan OS menghasilkan MAC-based UUID.<br><br>Perbaikan bug<br>• Peningkatan performa konfigurasi jaringan DHCP selama inisialisasi.<br>• Menghitung dengan benar checksum L4 pada lalu lintas masuk IPv6 ketika unit transmisi maksimum (MTU) melebihi 4K.<br>• Peningkatan umum untuk stabilitas driver dan perbaikan bug kecil. | 4 November 2019 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.1 | Perbaikan bug<br>• Mencegah penurunan paket TCP LSO yang sangat terfragmentasi, yang datang dari sistem operasi.<br>• Menangani protokol Encapsulating Security Payload (ESP) dengan benar di dalam IPSec pada jaringan IPv6. | 16 September 2019 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.0 | ENA Windows driver v2.1 memperkenalkan kapabilitas perangkat ENA baru, memberikan peningkatan performa, menambahkan fitur baru, dan memasukkan beberapa peningkatan stabilitas.<br><br>Fitur baru<br><br>• Menggunakan kunci registri Windows standar untuk konfigurasi frame Jumbo.<br>• Izinkan pengaturan ID VLAN melalui GUI properti driver ENA.<br>• Alur Pemulihan yang Ditingkatkan<br>  • Mekanisme identifikasi kegagalan yang ditingkatkan.<br>  • Menambahkan dukungan untuk parameter pemulihan yang bisa disesuaikan.<br>• Mendukung hingga 32 I/O antrian untuk instans EC2 baru yang memiliki lebih dari 8 vCPU.<br>• ~90% pengurangan jejak memori driver.<br><br>Optimalisasi performa<br><br>• Penurunan latensi jalur transmisi.<br>• Dukungan untuk menerima checksum offload.<br>• Pengoptimalan performa untuk sistem dengan beban berat (penggunaan mekanisme penguncian yang dioptimalkan). | 1 Juli 2019 |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| | • Penyempurnaan lebih lanjut untuk mengurangi pemakaian CPU dan meningkatkan responsivitas sistem saat diberi beban.<br><br>Perbaikan bug<br>• Memperbaiki crash karena penguraian yang tidak valid dari header Tx yang tidak berdekatan.<br>• Memperbaiki crash driver v1.5 selama pelepasan antarmuka jaringan elastis pada instans Bare Metal.<br>• Memperbaiki error perhitungan checksum pseudo-he ader LSO di atas IPv6.<br>• Memperbaiki potensi kebocoran sumber daya memori setelah kegagalan inisialisasi.<br>• N TCP/UDP onaktifkan checksum offload untuk fragmen IPv4.<br>• Memperbaiki konfigurasi VLAN. VLAN salah dinonakti fkan ketika hanya prioritas VLAN yang seharusnya dinonaktifkan.<br>• Mengaktifkan penguraian yang benar dari pesan driver kustom oleh pelihat peristiwa.<br>• Memperbaiki kegagalan untuk menginisialisasi driver karena penanganan stempel waktu yang tidak valid.<br>• Memperbaiki kondisi race antara pemrosesan data dan penonaktifan perangkat ENA. | |

| Versi driver | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.5.0 | Pembaruan<br>• Peningkatan stabilitas dan perbaikan performa.<br>• Buffer Terima sekarang dapat dikonfigurasi hingga nilai 8192 di Properti Lanjutan ENA NIC.<br>• Buffer Terima Default adalah 1k. | 4 Oktober 2018 |
| 1.2.3 | Mencakup perbaikan keandalan dan menyatukan dukungan untuk Windows Server 2008 R2 melalui Windows Server 2016. | 13 Februari 2018 |
| 1.0.8 | Rilis awal. Disertakan dalam AMI untuk Windows Server 2008 R2, Windows Server 2012 RTM, Windows Server 2012 R2, dan Windows Server 2016. | Juli 2016 |

Berlangganan pemberitahuan rilis driver ENA Windows dari Amazon SNS

Amazon SNS dapat memberi Anda notifikasi saat EC2 Windows Drivers versi baru dirilis. Gunakan prosedur berikut untuk berlangganan notifikasi ini.

Berlangganan notifikasi EC2

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di bilah navigasi, ubah Wilayah menjadi AS Timur (Virginia Utara), jika perlu. Anda harus memilih Wilayah ini karena notifikasi SNS langganan Anda ada di Wilayah ini.
3. Di panel navigasi, pilih Langganan.
4. Pilih Buat langganan.
5. Di kotak dialog Create subscription (Buat langganan), lakukan hal berikut:
    a. Untuk TopicARN, salin Amazon Resource Name (ARN) berikut:

arn: aws:sns:us-timur- 1:801119661308: ec2-windows-driver

b. Untuk Protokol, pilih `Email`.

c. Untuk  Endpoint, masukkan alamat email tempat Anda ingin pemberitahuan dikirim.

d. Pilih Buat langganan.

6. Anda akan menerima email konfirmasi. Buka email dan ikuti petunjuk untuk menyelesaikan langganan Anda.

Setiap kali driver EC2 Windows baru dirilis, kami mengirimkan notifikasi ke pelanggan. Jika Anda tidak ingin lagi menerima notifikasi ini, gunakan prosedur berikut untuk berhenti berlangganan.

Berhenti berlangganan dari notifikasi driver Windows Amazon EC2

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di panel navigasi, pilih Berlangganan.
3. Pilih kotak centang untuk langganan lalu pilih Tindakan, Hapus langganan. Ketika diminta konfirmasi, pilih Hapus.

## Driver paravirtual untuk instans Windows

AMI Windows berisi seperangkat driver untuk mengizinkan akses ke perangkat keras tervirtualisasi. Driver ini digunakan oleh Amazon EC2 untuk memetakan penyimpanan instans dan volume Amazon EBS ke perangkat mereka. Tabel berikut menunjukkan perbedaan utama antara driver yang berbeda.

| | Red Hat PV | Citrix PV | AWS PV |
|---|---|---|---|
| Tipe instans | Tidak didukung untuk semua tipe instans. Jika Anda menentukan tipe instans yang tidak didukung, instans tersebut akan mengalami gangguan. | Didukung untuk tipe instans Xen. | Didukung untuk tipe instans Xen. |
| Volume terlampir | Mendukung hingga 16 volume terlampir. | Mendukung lebih dari 16 volume terlampir. | Mendukung lebih dari |

| | Red Hat PV | Citrix PV | AWS PV |
|---|---|---|---|
| | | | 16 volume terlampir. |

| | Red Hat PV | Citrix PV | AWS PV |
|---|---|---|---|
| Jaringan | Pengemudi memiliki masalah yang diketahui saat koneksi jaringan disetel ulang saat beban tinggi; misalnya, transfer file FTP yang cepat. | | Pengemudi secara otomatis mengonfigurasi bingkai jumbo pada adaptor jaringan ketika pada tipe instans yang kompatibel. Ketika instance berada dalam grup penempatan cluster, ini menawarkan kinerja jaringan yang lebih baik antara instans yang berada dalam grup penempatan cluster. Untuk informasi selengkapnya, lihat Grup penempata |

| | Red Hat PV | Citrix PV | AWS PV |
|---|---|---|---|
| | | | n untuk instans Amazon EC2 Anda. |

Tabel berikut menunjukkan driver PV mana yang harus Anda jalankan di setiap versi Windows Server di Amazon EC2.

| Versi Windows Server | Versi driver PV |
|---|---|
| Windows Server 2025 | Tidak didukung |
| Windows Server 2022 | AWS PV versi terbaru |
| Windows Server 2019 | AWS PV versi terbaru |
| Windows Server 2016 | AWS PV versi terbaru |
| Windows Server 2012 R2 | AWS Versi PV 8.4.3 |
| Windows Server 2012 | AWS Versi PV 8.4.3 |
| Windows Server 2008 R2 | AWS Versi PV 8.3.5 |
| Windows Server 2008 | Citrix PV 5.9 |
| Windows Server 2003 | Citrix PV 5.9 |

Daftar Isi


- AWS Pengemudi PV
- Driver Citrix PV
- Driver Red Hat PV
- Berlangganan notifikasi
- Tingkatkan driver PV pada instans Windows EC2
- Memecahkan masalah driver PV pada instans Windows

## AWS Pengemudi PV

Driver AWS PV disimpan dalam `%ProgramFiles%\Amazon\Xentools` direktori. Direktori ini juga berisi simbol publik dan alat baris perintah,`xenstore_client.exe`, yang memungkinkan Anda untuk mengakses entri di XenStore. Misalnya, PowerShell perintah berikut mengembalikan waktu saat ini dari Hypervisor:

```
PS C:\> [DateTime]::FromFileTimeUTC((gwmi -n root\wmi -cl
 AWSXenStoreBase).XenTime).ToString("hh:mm:ss")
11:17:00
```

Kompon AWS en driver PV terdaftar di registri Windows di bawah`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`. Komponen driver tersebut adalah sebagai berikut: xenbus, xeniface, xennet, xenvbd, dan xenvif.

AWS Driver PV juga memiliki layanan Windows bernama LiteAgent, yang berjalan dalam mode pengguna. Ini menangani tugas-tugas seperti shutdown dan restart peristiwa dari AWS API pada instans generasi Xen. Anda dapat mengakses dan mengelola layanan dengan menjalankan `Services.msc` dari baris perintah. Saat berjalan pada instans generasi Nitro, driver AWS PV tidak digunakan dan LiteAgent layanan akan berhenti sendiri mulai dengan driver versi 8.2.4. Memperbarui ke driver AWS PV terbaru juga memperbarui LiteAgent dan meningkatkan keandalan pada semua generasi instans.

### Instal yang terbaru AWS Pengemudi PV

AMI Amazon Windows berisi seperangkat driver untuk mengizinkan akses ke perangkat keras virtual. Driver ini digunakan oleh Amazon EC2 untuk memetakan penyimpanan instans dan volume Amazon EBS ke perangkat mereka. Kami menyarankan Anda menginstal driver terbaru untuk meningkatkan stabilitas dan performa instans EC2 Windows Anda.

#### Opsi instalasi

- Gunakan AWS Systems Manager untuk memperbarui driver PV secara otomatis. Untuk informasi selengkapnya, lihat Panduan: Memperbarui Driver PV Secara Otomatis pada Instans Windows EC2 di Panduan AWS Systems Manager Pengguna.
- Unduh paket driver dan jalankan program instal secara manual. Untuk informasi tentang mengunduh dan menginstal driver AWS PV, atau memutakhirkan kontroler domain, lihat Tingkatkan instans Windows Server (AWS Upgrade PV) secara manual.

Untuk persyaratan sistem untuk menginstal paket AWS driver PV, lihatPersyaratan sistem untuk AWS Paket driver PV.

AWS Riwayat paket driver PV

Tabel berikut menunjukkan perubahan pada driver AWS PV untuk setiap rilis driver.

| Versi paket | Detail | Tanggal rilis |
|---|---|---|
| 8.6.1 8.6.1 | • Diperbarui dependensi runtime dalam paket penginstal. | Juli 21, 2026 |
| 8.6.0 | • Perbaikan stabilitas untuk XenStore interaksi. | Mei 27, 2025 |
| 8.5.0 | • Perbaikan stabilitas untuk mengatasi kasus crash yang jarang terjadi selama pelepasan perangkat jaringan.<br>• Perbaikan stabilitas untuk mengatasi kasus crash yang jarang terjadi selama pelepasan volume EBS.<br>• Memperbaiki bug di penginstal paket.<br>• Memperbarui penginstal PV untuk digunakan`Pnputil`. | Oktober 31, 2024 |
| 8.4.3 8.4.3 | Memperbaiki bug di penginstal paket untuk meningkatkan pengalaman pemutakhiran. Ini adalah versi terakhir yang dapat berjalan pada Windows Server 2012 dan 2012 R2. Versi ini tersedia untuk diunduh, namun tidak lagi didukung sejak Windows Server 2012 dan 2012 R2 telah mencapai akhir dukungan. | Januari 24, 2023 |
| 8.4.2 | Perbaikan stabilitas untuk mengatasi kondisi balapan. | 13 April 2022 |
| 8.4.1 | Penginstal paket yang ditingkatkan. | 7 Januari 2022 |
| 8.4.0 | • Perbaikan stabilitas untuk mengatasi kasus langka terjebak disk IO.<br>• Perbaikan stabilitas untuk mengatasi kasus crash yang jarang terjadi selama pelepasan volume EBS.<br>• Menambahkan fitur untuk mendistribusikan beban di banyak inti untuk beban kerja yang memanfaatkan lebih dari 20.000 | 2 Maret 2021 |

| Versi paket | Detail | Tanggal rilis |
| --- | --- | --- |
| | IOPS dan mengalami degradasi akibat hambatan. Untuk mengaktifkan fitur ini, lihat Beban kerja yang memanfaatkan lebih dari 20.000 disk IOPS mengalami degradasi karena kemacetan CPU. | |
| 8.3.5 8.3.5 | Penginstal paket yang ditingkatkan.<br><br>Ini adalah versi terakhir yang dapat berjalan pada Windows Server 2008 R2. Versi ini tersedia untuk diunduh tetapi tidak lagi didukung. Windows Server 2008 R2 telah mencapai akhir masa pakai, dan tidak lagi didukung oleh Microsoft. | 7 Januari 2022 |
| 8.3.4 | Peningkatan keandalan lampiran perangkat jaringan. | 4 Agustus 2020 |
| 8.3.3 | • Perbarui XenStore-facing komponen untuk mencegah pemeriksaan bug selama jalur penanganan kesalahan.<br>• Perbarui ke komponen penyimpanan untuk menghindari kerusakan ketika SRB tidak valid dikirimkan.<br><br>Untuk memperbarui driver ini pada instans Windows Server 2008 R2, Anda harus terlebih dahulu memverifikasi bahwa patch yang sesuai telah diinstal untuk mengatasi Microsoft Security Advisory berikut ini: Microsoft Security Advisory 3033929. | 4 Februari 2020 |
| 8.3.2 | Keandalan yang ditingkatkan dari komponen jaringan. | 30 Juli 2019 |
| 8.3.1 | Peningkatan kinerja dan ketahanan komponen penyimpanan. | 12 Juni 2019 |
| 8.2.7 | Peningkatan efisiensi untuk mendukung migrasi ke tipe instans generasi terbaru. | 20 Mei 2019 |
| 8.2.6 | Peningkatan efisiensi jalur dump kecelakaan. | Januari 15, 2019 |

| Versi paket | Detail | Tanggal rilis |
| --- | --- | --- |
| 8.2.5 | Peningkatan keamanan tambahan.<br><br>PowerShell installer sekarang tersedia dalam paket. | 12 Desember 2018 |
| 8.2.4 | Peningkatan keandalan. | 2 Oktober 2018 |
| 8.2.3 | Perbaikan bug dan peningkatan performa.<br><br>Laporkan ID volume EBS sebagai nomor seri disk untuk volume EBS. Hal ini memungkinkan skenario klaster seperti S2D. | 29 Mei 2018 |
| 8.2.1 | Peningkatan kinerja jaringan dan penyimpanan ditambah beberapa perbaikan ketahanan.<br><br>Untuk memverifikasi bahwa versi ini telah diinstal, lihat nilai registri Windows berikut ini: `HKLM\Software\Amazon\PVDriver\Version 8.2.1` . | 8 Maret 2018 |
| 7.4.3 | Dukungan tambahan untuk Windows Server 2016.<br><br>Perbaikan stabilitas untuk semua versi OS Windows yang didukung.<br><br>*AWS Tanda tangan driver PV versi 7.4.3 berakhir pada 29 Maret 2019. Kami merekomendasikan memperbarui ke driver AWS PV terbaru. | 18 November 2016 |
| 7.4.2 | Perbaikan stabilitas untuk dukungan tipe instans X1. | Agustus 2, 2016 |
| 7.4.1 | • Peningkatan kinerja dalam driver AWS Penyimpanan PV.<br>• Perbaikan stabilitas AWS di driver Penyimpanan PV: Memperbaiki masalah di mana instance mengalami kerusakan sistem dengan kode pemeriksaan bug 0x0000DEAD.<br>• Perbaikan stabilitas di AWS driver Jaringan PV.<br>• Menambahkan dukungan untuk Windows Server 2008R2. | 12 Juli 2016 |

| Versi paket | Detail | Tanggal rilis |
| --- | --- | --- |
| 7.3.2 | • Peningkatan logging dan diagnostik.<br>• Perbaikan stabilitas di AWS driver Penyimpanan PV. Dalam beberapa kasus, disk mungkin tidak muncul di Windows setelah menyambungkan kembali disk ke instans.<br>• Menambahkan dukungan untuk Windows Server 2012. | 24 Juni 2015 |
| 7.3.1 | Pembaruan TRIM: Perbaikan terkait dengan permintaan TRIM. Perbaikan ini menstabilkan instans dan meningkatkan kinerja instans saat mengelola permintaan TRIM dalam jumlah besar. | |
| 7.3.0 | Dukungan TRIM: Driver AWS PV sekarang mengirimkan permintaan TRIM ke hypervisor. Disk efemeral akan memproses permintaan TRIM dengan benar karena penyimpanan yang mendasarinya mendukung TRIM (SSD). Perhatikan bahwa EBS-based penyimpanan tidak mendukung TRIM pada Maret 2015. | |
| 7.2.5 | • Perbaikan stabilitas AWS di driver Penyimpanan PV: Dalam beberapa kasus, AWS driver PV dapat menghilangkan referensi memori yang tidak valid dan menyebabkan kegagalan sistem.<br>• Perbaikan stabilitas saat menghasilkan crash dump: Dalam beberapa kasus, AWS pengemudi PV bisa terjebak dalam kondisi balapan saat menulis crash dump. Sebelum rilis ini, masalah hanya dapat diatasi dengan memaksa driver untuk berhenti dan memulai ulang yang dapat menyebabkan hilangnya timbunan memori. | |
| 7.2.4 | Persistensi ID Perangkat: Perbaikan driver ini menutupi ID perangkat PCI platform dan memaksa sistem untuk selalu memunculkan ID perangkat yang sama, meskipun instans dipindahkan. Secara lebih umum, perbaikan memengaruhi cara hypervisor menampilkan perangkat virtual. Perbaikan juga mencakup modifikasi pada co-installer untuk driver AWS PV sehingga sistem tetap mempertahankan perangkat virtual yang dipetakan. | |

| Versi paket | Detail | Tanggal rilis |
| --- | --- | --- |
| 7.2.2 | • Muat driver AWS PV dalam mode Mode Pemulihan Layanan Direktori (DSRM): Mode Pemulihan Layanan Direktori adalah opsi boot mode aman untuk pengontrol domain Windows Server.<br>• Menjaga ID perangkat tetap ada ketika perangkat adaptor jaringan virtual terpasang kembali: Perbaikan ini memaksa sistem untuk memeriksa pemetaan alamat MAC dan mempertahankan ID perangkat. Perbaikan ini memastikan bahwa adaptor mempertahankan pengaturan statisnya jika adaptor dipasang kembali. | |
| 7.2.1 | • Jalankan di mode aman: Memperbaiki masalah di mana driver tidak mau memuat dalam mode aman. Sebelumnya AWS PV Drivers hanya akan membuat instance dalam sistem yang berjalan normal.<br>• Tambahkan disk ke Microsoft Windows Storage Pools: Sebelumnya kami menyintesis kueri halaman 83. Perbaikan menonaktifkan dukungan halaman 83. Perhatikan bahwa ini tidak memengaruhi kolam penyimpanan yang digunakan di lingkungan klaster karena disk PV bukan disk klaster yang valid. | |
| 7.2.0 | Basis: Versi AWS dasar PV. | |

## Driver Citrix PV

Driver Citrix PV disimpan di driektori `%ProgramFiles%\Citrix\XenTools` (instans 32-bit) atau `%ProgramFiles(x86)%\Citrix\XenTools` (64-bit instans).

Komponen driver Citrix PV tercantum dalam registri Windows di bawah `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services`. Komponen driver tersebut adalah sebagai berikut: xenevtchn, xeniface, xennet, Xennet6, xensvc, xenvbd, and xenvif.

Citrix juga memiliki komponen driver bernama XenGuestAgent, yang berjalan sebagai layanan Windows. Layanan ini menangani tugas-tugas seperti mematikan dan memulai ulang peristiwa dari

API. Anda dapat mengakses dan mengelola layanan dengan menjalankan `Services.msc` dari baris perintah.

Jika Anda mengalami kesalahan jaringan saat melakukan beban kerja tertentu, Anda mungkin perlu menonaktifkan fitur pemindahan TCP untuk driver Citrix PV. Untuk informasi selengkapnya, lihat Pemindahan TCP.

## Driver Red Hat PV

Driver Red Hat didukung untuk instans lama, tetapi tidak direkomendasikan pada instans yang lebih baru dengan RAM lebih dari 12GB karena keterbatasan driver. Instans dengan RAM lebih dari 12GB yang menjalankan driver Red Hat dapat gagal boot dan menjadi tidak dapat diakses. Kami merekomendasikan upgrade driver Red Hat ke driver Citrix PV, dan kemudian tingkatkan driver Citrix PV ke driver P AWS V.

File sumber untuk driver Red Hat ada di direktori `%ProgramFiles%\RedHat` (instance 32-bit) atau `%ProgramFiles(x86)%\RedHat` (64-bit instance). Kedua driver tersebut adalah `rhelnet` driver jaringan Red Hat Paravirtualized`rhelscsi`, dan driver miniport Red Hat SCSI.

## Berlangganan notifikasi

Amazon SNS dapat memberi Anda notifikasi saat EC2 Windows Drivers versi baru dirilis. Anda dapat berlangganan notifikasi ini.

> **Note**
>
> Anda harus menentukan Wilayah tempat topik SNS dibuat.

Setiap kali driver EC2 Windows baru dirilis, kami mengirimkan notifikasi ke pelanggan. Jika Anda tidak lagi ingin menerima pemberitahuan ini, Anda dapat berhenti berlangganan. Untuk informasi selengkapnya, lihat Menghapus topik SNS dan langganan.

Console

Untuk berlangganan notifikasi

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di bilah navigasi, ubah Wilayah menjadi AS Timur (Virginia Utara), jika perlu. Anda harus memilih Wilayah ini karena notifikasi SNS langganan Anda ada di Wilayah ini.

3. Di panel navigasi, pilih Langganan.
4. Pilih Buat langganan.
5. Di kotak dialog Create subscription (Buat langganan), lakukan hal berikut:
   a. Untuk TopicARN, salin Amazon Resource Name (ARN) berikut:

      arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers
   b. Untuk Protokol, pilih `Email`.
   c. Untuk Titik Akhir, ketik alamat email yang bisa Anda gunakan untuk menerima notifikasi.
   d. Pilih Buat langganan.
6. Anda akan menerima email konfirmasi. Buka email dan ikuti petunjuk untuk menyelesaikan langganan Anda.

AWS CLI

Untuk berlangganan notifikasi

Gunakan perintah berikut ini.

```
aws sns subscribe \
    --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers \
    --region us-east-1 \
    --protocol email \
    --notification-endpoint YourUserName@YourDomainName.ext
```

PowerShell

Untuk berlangganan notifikasi

Gunakan perintah berikut ini.

```
Connect-SNSNotification `
    -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers' `
    -Region us-east-1 `
    -Protocol email `
    -Endpoint "YourUserName@YourDomainName.ext"
```

# Tingkatkan driver PV pada instans Windows EC2

Kami menyarankan Anda menginstal driver PV terbaru untuk meningkatkan stabilitas dan performa instans EC2 Windows Anda. Petunjuk di halaman ini membantu Anda mengunduh paket driver dan menjalankan program penginstalan.

Untuk memverifikasi driver mana yang digunakan instans Windows Anda

Buka  Device Manager  dan lihat Adap  tor Jaringan. Periksa apakah driver PV adalah salah satu dari berikut ini:

- AWS Perangkat Jaringan PV
- Adaptor Ethernet Citrix PV
- Driver Red Hat PV NIC

Daftar Isi


- Persyaratan sistem untuk AWS Paket driver PV
- Tingkatkan instans Windows Server (AWS Upgrade PV) dengan Distributor
- Tingkatkan instans Windows Server (AWS Upgrade PV) secara manual
- Tingkatkan pengontrol domain (AWS Peningkatan PV)
- Tingkatkan instans Windows Server 2008 dan 2008 R2 (Red Hat ke Citrix PV upgrade)
- Mutakhirkan layanan agen tamu Citrix Xen Anda


## Persyaratan sistem untuk AWS Paket driver PV

Persyaratan berikut berlaku untuk paket driver AWS PV apakah Anda menginstalnya melalui Manajer Sistem atau secara manual.

- Versi Windows Server yang didukung versi paket. Versi Windows Server yang lebih lama memerlukan versi paket yang lebih lama. Untuk versi paket yang akan digunakan dengan sistem operasi Anda, lihatDriver paravirtual untuk instans Windows.
- Versi minimum .NET Framework yang diperlukan oleh penginstal paket, seperti yang ditunjukkan tabel berikut.

| AWS Versi paket driver PV | Minimum.NET Framework versi |
|---|---|
| Terbaru | 4.7.2 |
| 8.4.3 | 4.5 |
| 8.3.5 | 4.5 |

Tingkatkan instans Windows Server (AWS Upgrade PV) dengan Distributor

Anda dapat menggunakan Distributor, kemampuan AWS Systems Manager, untuk menginstal atau meningkatkan paket AWS driver PV. Instalasi atau peningkatan dapat dilakukan satu kali, atau Anda dapat menginstal atau memperbaruinya sesuai jadwal. `In-place update`Opsi untuk Jenis Instalasi tidak didukung untuk paket Distributor ini.

**Important**

Jika instans Anda adalah kontroler domain, lihat Tingkatkan pengontrol domain (AWS Peningkatan PV). Proses pemutakhiran versi untuk instans kontroler domain berbeda dari edisi standar Windows.

1. Kami menyarankan Anda membuat cadangan jika Anda perlu mengembalikan perubahan Anda.

   **Tip**

   Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Otomasi Manajer Sistem untuk membuat AMI menggunakan `AWS-CreateImage` runbook. Untuk informasi selengkapnya, lihat AWS-CreateImage di Panduan Pengguna referensi runbook AWS Systems Manager Otomasi.

   a. Ketika Anda menghentikan suatu instans, data pada setiap instans volume penyimpanan akan dihapus. Sebelum Anda menghentikan sebuah instans, pastikan bahwa Anda telah menyalin data apa pun yang Anda perlukan dari volume penyimpanan instans ke penyimpanan persisten, seperti Amazon EBS atau Amazon S3.

   b. Di panel navigasi, pilih Contoh.

c. Pilih instans yang memerlukan pemutakhiran driver, dan pilih Status instans, Hentikan instans.

d. Setelah instans dihentikan, pilih instans, pilih Tindakan, lalu Gambar dan templat, lalu pilih Buat gambar.

e. Pilih Status instans, Mulai instans.

2. Hubungkan ke instans menggunakan Desktop Jarak Jauh. Untuk informasi selengkapnya, lihat the section called “Hubungkan menggunakan klien RDP”.

3. Kami menyarankan Anda untuk membuat semua disk non-sistem offline dan mencatat setiap pemetaan huruf drive ke disk sekunder di Manajemen Disk sebelum Anda melakukan pemutakhiran ini. Langkah ini tidak diperlukan jika Anda melakukan pembaruan driver AWS PV di tempat. Kami juga merekomendasikan pengaturan layanan yang tidak penting ke start-up Manual di konsol Layanan.

4. Untuk petunjuk tentang cara menginstal atau meningkatkan paket driver AWS PV menggunakan Distributor, lihat prosedur di Menginstal atau memperbarui paket di Panduan AWS Systems Manager Pengguna.

5. Untuk Nama, pilih AwspvDriver.

6. Untuk Jenis Instalasi, pilih Uninstall dan instal ulang.

7. Konfigurasikan parameter lain untuk paket seperlunya dan jalankan instalasi atau upgrade menggunakan prosedur yang direferensikan diStep 4.

   Setelah menjalankan paket Distributor, instans secara otomatis melakukan reboot dan kemudian memutakhirkan driver. Instans tidak akan tersedia hingga 15 menit.

8. Setelah pemutakhiran selesai, dan instans melewati kedua pemeriksaan kesehatan di konsol Amazon EC2, verifikasi bahwa driver baru diinstal dengan menghubungkan ke instans menggunakan Remote Desktop.

9. Setelah Anda terhubung, jalankan PowerShell perintah berikut:

```
Get-ItemProperty HKLM:\SOFTWARE\Amazon\PVDriver
```

10. Pastikan versi driver sama dengan versi terbaru yang terdaftar pada tabel Riwayat Versi Driver. Untuk informasi selengkapnya, lihat AWS Riwayat paket driver PV Buka Manajemen Disk untuk meninjau volume sekunder offline dan membawanya online sesuai dengan huruf drive yang tercantum diStep 3.

Jika sebelumnya Anda menonaktifkan Pemindahan TCP menggunakan driver Netsh untuk Citrix PV, kami sarankan Anda mengaktifkan kembali fitur ini setelah memutakhirkan ke AWS driver PV. Masalah TCP Offloading dengan driver Citrix tidak ada di driver PV. AWS Akibatnya, TCP Offloading memberikan kinerja yang lebih baik dengan driver AWS PV.

Jika sebelumnya Anda menerapkan alamat IP statis atau konfigurasi DNS ke antarmuka jaringan, Anda mungkin perlu menerapkan kembali alamat IP statis atau konfigurasi DNS setelah memutakhirkan driver AWS PV.

Tingkatkan instans Windows Server (AWS Upgrade PV) secara manual

Gunakan prosedur berikut untuk melakukan upgrade driver AWS PV di tempat, atau untuk memutakhirkan dari driver Citrix PV ke driver AWS PV pada Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, Windows Server 2019, atau Windows Server 2022. Upgrade ini tidak tersedia untuk driver Red Hat, atau untuk versi Windows Server lainnya.

> **Important**
>
> Jika instans Anda adalah kontroler domain, lihat Tingkatkan pengontrol domain (AWS Peningkatan PV). Proses pemutakhiran versi untuk instans kontroler domain berbeda dari edisi standar Windows.

Untuk meng-upgrade AWS Driver PV secara manual

1. Kami menyarankan Anda membuat cadangan jika Anda perlu mengembalikan perubahan Anda.

   > **Tip**
   >
   > Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Otomasi Manajer Sistem untuk membuat AMI menggunakan `AWS-CreateImage` runbook. Untuk informasi selengkapnya, lihat AWS-CreateImage di Panduan Pengguna referensi runbook AWS Systems Manager Otomasi.

   a. Ketika Anda menghentikan suatu instans, data pada setiap instans volume penyimpanan akan dihapus. Sebelum Anda menghentikan sebuah instans, pastikan bahwa Anda

telah menyalin data apa pun yang Anda perlukan dari volume penyimpanan instans ke penyimpanan persisten, seperti Amazon EBS atau Amazon S3.

b. Di panel navigasi, pilih Contoh.

c. Pilih instans yang memerlukan pemutakhiran driver, dan pilih Status instans, Hentikan instans.

d. Setelah instans dihentikan, pilih instans, pilih Tindakan, lalu Gambar dan templat, lalu pilih Buat gambar.

e. Pilih Status instans, Mulai instans.

2. Hubungkan ke instans menggunakan Desktop Jarak Jauh.

3. Kami menyarankan Anda untuk membuat semua disk non-sistem offline dan mencatat setiap pemetaan huruf drive ke disk sekunder di Manajemen Disk sebelum Anda melakukan pemutakhiran ini. Langkah ini tidak diperlukan jika Anda melakukan pembaruan driver AWS PV di tempat. Kami juga merekomendasikan pengaturan layanan yang tidak penting ke start-up Manual di konsol Layanan.

4. Unduh driver ke instance Anda menggunakan salah satu opsi berikut:

   - Browser - Unduh paket driver terbaru ke instance dan ekstrak arsip zip.
   - PowerShell— Jalankan perintah berikut:

     ```
     Invoke-WebRequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/AWSPV/
     Latest/AWSPVDriver.zip -outfile $env:USERPROFILE\pv_driver.zip
     Expand-Archive $env:userprofile\pv_driver.zip -DestinationPath $env:userprofile
     \pv_drivers
     ```

     Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:

     ```
     [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
     ```

5. Jalankan `AWSPVDriverSetup.msi`.

Setelah menjalankan MSI, instans secara otomatis melakukan boot ulang kemudian memutakhirkan driver. Instance tidak akan tersedia hingga 15 menit. Setelah pemutakhiran selesai dan instans melewati kedua pemeriksaan kesehatan di konsol Amazon EC2, Anda dapat memverifikasi bahwa

driver baru diinstal dengan menghubungkan ke instans menggunakan Remote Desktop dan kemudian menjalankan PowerShell perintah berikut:

```
Get-ItemProperty HKLM:\SOFTWARE\Amazon\PVDriver
```

Pastikan versi driver sama dengan versi terbaru yang terdaftar pada tabel Riwayat Versi Driver. Untuk informasi selengkapnya, lihat AWS Riwayat paket driver PV Buka Manajemen Disk untuk meninjau volume sekunder offline dan membawanya online sesuai dengan huruf drive yang tercantum diStep 3.

Jika sebelumnya Anda menonaktifkan Pemindahan TCP menggunakan driver Netsh untuk Citrix PV, kami sarankan Anda mengaktifkan kembali fitur ini setelah memutakhirkan ke AWS driver PV. Masalah TCP Offloading dengan driver Citrix tidak ada di driver PV. AWS Akibatnya, TCP Offloading memberikan kinerja yang lebih baik dengan driver AWS PV.

Jika sebelumnya Anda menerapkan alamat IP statis atau konfigurasi DNS ke antarmuka jaringan, Anda mungkin perlu menerapkan kembali alamat IP statis atau konfigurasi DNS setelah memutakhirkan driver AWS PV.

Tingkatkan pengontrol domain (AWS Peningkatan PV)

Gunakan prosedur berikut pada pengontrol domain untuk melakukan upgrade driver AWS PV di tempat, atau untuk memutakhirkan dari driver Citrix PV ke driver P AWS V. Untuk memastikan bahwa peran FSMO Anda tetap beroperasi selama pemutakhiran, kami sarankan Anda mentransfer peran tersebut ke pengontrol domain lain sebelum memulai peningkatan. Untuk informasi selengkapnya, lihat Cara melihat dan mentransfer peran FSMO di situs web Microsoft Learn.

Untuk meningkatkan kontroler domain

1. Kami menyarankan agar Anda membuat cadangan kontroler domain jika Anda perlu mengembalikan perubahan Anda. Menggunakan AMI sebagai cadangan tidak didukung. Untuk informasi selengkapnya, lihat Pertimbangan pencadangan dan pemulihan dalam dokumentasi Microsoft.
2. Jalankan perintah berikut untuk mengonfigurasi Windows agar booting ke Mode Pemulihan Layanan Direktori (DSRM).

> **Warning**
>
> Sebelum menjalankan perintah ini, konfirmasikan bahwa Anda mengetahui kata sandi DSRM. Anda akan memerlukan informasi ini agar Anda dapat masuk ke instans setelah pemutakhiran versi selesai dan instans di-boot ulang secara otomatis.

```
bcdedit /set {default} safeboot dsrepair
```

PowerShell:

```
PS C:\> bcdedit /set "{default}" safeboot dsrepair
```

Sistem harus boot ke DSRM karena utilitas upgrade menghapus driver penyimpanan Citrix PV sehingga dapat menginstal driver AWS PV. Oleh karena itu kami menyarankan untuk mencatat pemetaan huruf dan folder drive apa pun ke disk sekunder di Manajemen Disk. Jika driver penyimpanan Citrix PV tidak ada, drive sekunder tidak terdeteksi. Kontroler domain yang menggunakan folder NTDS di drive sekunder tidak akan bisa di-boot karena disk sekunder tidak terdeteksi.

> **Warning**
>
> Setelah Anda menjalankan perintah ini, jangan boot ulang sistem secara manual. Sistem tidak dapat dijangkau karena driver Citrix PV tidak mendukung DSRM.

3. Jalankan perintah berikut untuk ditambahkan **`DisableDCCheck`** ke pendataan ini:

```
reg add HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck /t
 REG_SZ /d true
```

4. Unduh paket driver terbaru ke instance dan ekstrak arsip zip.
5. Jalankan `AWSPVDriverSetup.msi`.

   Setelah menjalankan MSI, instans secara otomatis melakukan boot ulang kemudian memutakhirkan driver. Instans tidak akan tersedia hingga 15 menit.

6. Setelah pemutakhiran selesai dan instans melewati kedua pemeriksaan kondisi di konsol Amazon EC2, hubungkan ke instans menggunakan Remote Desktop. Buka Manajemen Disk untuk meninjau volume sekunder offline dan membuatnya online sesuai dengan huruf drive dan pemetaan folder yang disebutkan sebelumnya.

   Anda harus terhubung ke instance dengan menentukan nama pengguna dalam format berikut nama host\ administrator. Sebagai contoh, Win2k12TestBox \administrator.
7. Jalankan perintah berikut untuk menghapus konfigurasi boot DSRM:

   ```
   bcdedit /deletevalue safeboot
   ```
8. Boot ulang instans.
9. Untuk menyelesaikan proses pemutakhiran, pastikan bahwa driver baru telah diinstal. Di Pengelola Perangkat, di bawah Kontroler Penyimpanan, temukan Adaptor Host Penyimpanan AWS PV. Pastikan versi driver sama dengan versi terbaru yang terdaftar pada tabel Riwayat Versi Driver. Untuk informasi selengkapnya, lihat AWS Riwayat paket driver PV.
10. Jalankan perintah berikut untuk menghapus **`DisableDCCheck`** dari pendataan ini:

    ```
    reg delete HKLM\SOFTWARE\Wow6432Node\Amazon\AWSPVDriverSetup /v DisableDCCheck
    ```

> **Note**
>
> Jika sebelumnya Anda menonaktifkan Pemindahan TCP menggunakan driver Netsh untuk Citrix PV, kami sarankan Anda mengaktifkan kembali fitur ini setelah memutakhirkan ke AWS Driver PV. Masalah TCP Offloading dengan driver Citrix tidak ada di driver PV. AWS Akibatnya, TCP Offloading memberikan kinerja yang lebih baik dengan driver AWS PV.

Tingkatkan instans Windows Server 2008 dan 2008 R2 (Red Hat ke Citrix PV upgrade)

Sebelum Anda mulai memutakhirkan driver Red Hat Anda ke driver Citrix PV, pastikan Anda melakukan hal berikut:

- Instal versi terbaru dari layanan EC2Config. Untuk informasi selengkapnya, lihat Menginstal EC2Config versi terbaru.
- Pastikan bahwa Anda telah menginstal Windows PowerShell 3.0. Untuk memverifikasi versi yang telah Anda instal, jalankan perintah berikut di file PowerShell jendela:

```
PS C:\> $PSVersionTable.PSVersion
```

Windows PowerShell 3.0 dibundel dalam paket instalasi Windows Management Framework (WMF) versi 3.0. Jika Anda perlu menginstal Windows PowerShell 3.0, lihat Windows Management Framework 3.0 di Pusat Unduhan Microsoft.

- Cadangkan informasi penting Anda pada instans, atau buat AMI dari instans. Untuk informasi selengkapnya tentang cara membuat AMI, lihat Buat EBS-backed AMI Amazon.

> **Tip**
>
> Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Otomasi Manajer Sistem untuk membuat AMI menggunakan `AWS-CreateImage` runbook. Untuk informasi selengkapnya, lihat AWS-CreateImage di Panduan Pengguna referensi runbook AWS Systems Manager Otomasi.

Jika Anda membuat AMI, pastikan Anda melakukan hal berikut:

- Tuliskan kata sandi Anda.
- Jangan menjalankan alat Sysprep secara manual atau menggunakan file layanan EC2Config.
- Setel adaptor Ethernet Anda untuk mendapatkan alamat IP secara otomatis menggunakan DHCP.

Untuk meng-upgrade driver Red Hat

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal. Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat Hubungkan ke instans Windows Anda menggunakan RDP.
2. Dalam instans Anda, unduh paket pemutakhiran Citrix PV.
3. Ekstrak konten paket yang dimutakhirkan ke lokasi pilihan Anda.
4. Buka file Upgrade.bat. Jika Anda mendapatkan peringatan keamanan, pilih Jalankan.
5. Di kotak dialog Tingkatkan Driver, tinjau informasinya dan pilih Ya jika Anda siap untuk memulai peningkatan.
6. Di kotak dialog Red Hat Paravirtualized Xen Drivers for Windows uninstaller, pilih Ya untuk menghapus perangkat lunak Red Hat. Instans Anda akan di-boot ulang.

> **Note**
>
> Jika Anda tidak melihat kotak dialog uninstaller, pilih Red Hat Paravirtualize di taskbar Windows.



7. Periksa apakah instans telah di-boot ulang dan siap digunakan.

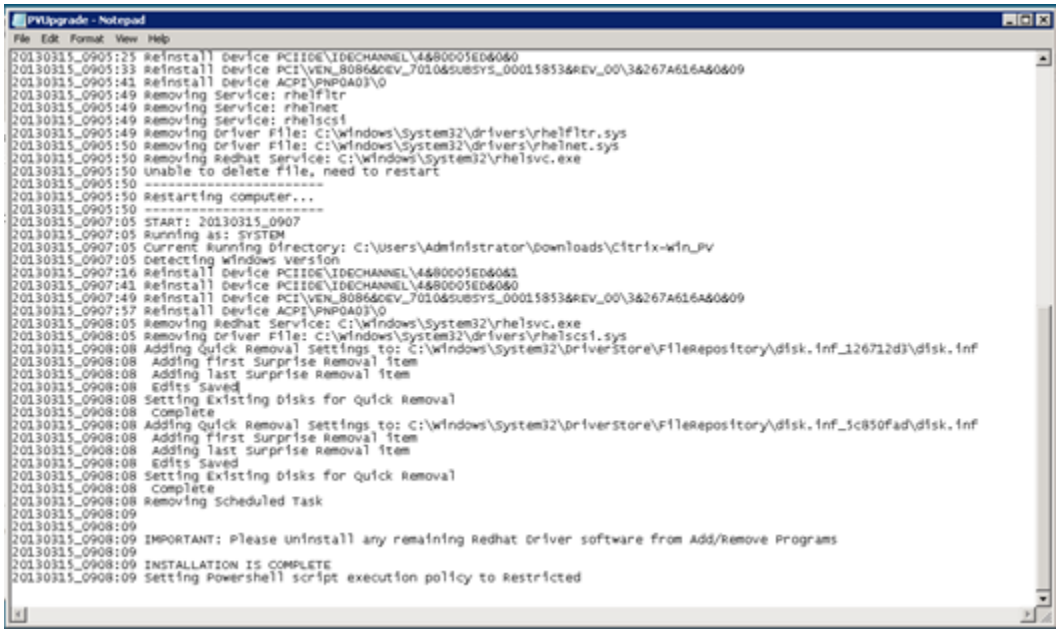
   a. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

   b. Pada halaman Instans, pilih Tindakan, lalu Pantau dan pecahkan masalah, lalu pilih Dapatkan log sistem.

   c. Operasi pemutakhiran harus memulai ulang server 3 atau 4 kali. Anda dapat melihat ini di file log dengan berapa kali `Windows is Ready to use` ditampilkan.

```
    Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
 Ec2Config service v2.1.9.0
  RedHat PV NIC Driver v1.3.10.0
2013/03/15 17:11:01Z: Waiting for meta-data accessibility...
2013/03/15 17:11:02Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
<Username>Administrator</Username>
<Password>
L79ThJPF8LyIL38IZht0FBrjet3vnT2csTiU/XGVMRCH7kQtBnznAnXrKd1sirXlxl9BwVMsd9b38jFJqv0lIUpgNNJRZoCDc7IbUw
</Password>
2013/03/15 17:11:30Z: Product activation was successful.
2013/03/15 17:11:32Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
 Ec2Config service v2.1.9.0
2013/03/15 21:04:24Z: There was an exception writing driver information to console: System.Exception:
   at Ec2Config.Service1.Go()
2013/03/15 21:04:35Z: Waiting for meta-data accessibility...
2013/03/15 21:04:40Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:05:08Z: Product activation was successful.
2013/03/15 21:05:09Z: Message: Windows is Ready to use
Microsoft Windows NT 6.0.6002 Service Pack 2 (en-US)
 Ec2Config service v2.1.9.0
  Citrix PV Ethernet Adapter v5.9.960.49119
2013/03/15 21:07:20Z: Waiting for meta-data accessibility...
2013/03/15 21:07:21Z: Meta-data is now available.
<RDPCERTIFICATE>
<THUMBPRINT>D6BFD64F21359516C781CA7DF2821C5EFC35648A</THUMBPRINT>
</RDPCERTIFICATE>
2013/03/15 21:07:27Z: Message: Windows is Ready to use
```

8. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
9. Tutup kotak dialog Red Hat Paravirtualized Xen Drivers untuk pelepas instalasi Windows.
10. Konfirmasikan bahwa penginstalan selesai. Arahkan ke folder `Citrix-WIN_PV` yang Anda ekstrak sebelumnya, buka file `PVUpgrade.log`, lalu centang teks `INSTALLATION IS COMPLETE`.




Mutakhirkan layanan agen tamu Citrix Xen Anda

Jika Anda menggunakan driver Citrix PV di Windows Server, Anda dapat memutakhirkan layanan agen tamu Citrix Xen. Layanan Windows menangani tugas-tugas seperti mematikan dan memulai ulang peristiwa dari API. Anda dapat menjalankan paket pemutakhiran ini di versi Windows Server apa pun, selama instans menjalankan driver Citrix PV.

> **Important**
>
> Untuk Windows Server 2008 R2 dan yang lebih baru, kami sarankan Anda memutakhirkan ke AWS driver PV yang menyertakan pembaruan Agen Tamu.

Sebelum Anda mulai memutakhirkan driver, pastikan Anda mencadangkan informasi penting pada instans, atau buat AMI dari instans. Untuk informasi selengkapnya tentang membuat AMI, lihat Buat EBS-backed AMI Amazon.

> **Tip**
>
> Alih-alih membuat AMI dari konsol Amazon EC2, Anda dapat menggunakan Otomasi Manajer Sistem untuk membuat AMI menggunakan `AWS-CreateImage` runbook. Untuk informasi

selengkapnya, lihat AWS-CreateImage di Panduan Pengguna referensi runbook AWS Systems Manager Otomasi.

Jika Anda membuat AMI, pastikan Anda melakukan hal berikut:

- Jangan aktifkan alat Sysprep di file layanan EC2Config.
- Tuliskan kata sandi Anda.
- Setel adaptor Ethernet Anda ke DHCP.

Untuk meningkatkan layanan agen tamu Citrix Xen Anda

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal. Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat Hubungkan ke instans Windows Anda menggunakan RDP.
2. Pada instans, unduh paket pemutakhiran Citrix.
3. Ekstrak konten paket yang dimutakhirkan ke lokasi pilihan Anda.
4. Buka file Upgrade.bat. Jika Anda mendapatkan peringatan keamanan, pilih Jalankan.
5. Di kotak dialog Tingkatkan Driver, tinjau informasinya dan pilih Ya jika Anda siap untuk memulai peningkatan.
6. Saat pemutakhiran selesai, file `PVUpgrade.log` akan terbuka dan berisi teks `UPGRADE IS COMPLETE`.
7. Booting ulang instans Anda.

## Memecahkan masalah driver PV pada instans Windows

Berikut ini adalah solusi untuk masalah yang mungkin Anda temui dengan image Amazon EC2 dan driver PV yang lebih lama.

Daftar Isi


- Windows Server 2012 R2 kehilangan konektivitas jaringan dan penyimpanan setelah boot ulang instans
- Pemindahan TCP
- Sinkronisasi waktu

- Beban kerja yang memanfaatkan lebih dari 20.000 disk IOPS mengalami degradasi karena kemacetan CPU

Windows Server 2012 R2 kehilangan konektivitas jaringan dan penyimpanan setelah boot ulang instans

> **Important**
>
> Masalah ini hanya terjadi dengan AMI yang tersedia sebelum September 2014.

Amazon Machine Image (AMI) Windows Server 2012 R2 yang tersedia sebelum 10 September 2014 dapat kehilangan konektivitas jaringan dan penyimpanan setelah boot ulang instans. Kesalahan dalam log Konsol Manajemen AWS sistem menyatakan: “Kesulitan mendeteksi detail driver PV untuk Output Konsol.” Hilangnya konektivitas disebabkan oleh fitur Plug and Play Cleanup. Fitur ini memindai dan menonaktifkan perangkat sistem yang tidak aktif setiap 30 hari. Fitur tersebut salah mengidentifikasi perangkat jaringan EC2 sebagai tidak aktif dan menghapusnya dari sistem. Jika ini terjadi, instans kehilangan konektivitas jaringan setelah boot ulang.

Untuk sistem yang Anda curigai dapat terpengaruh oleh masalah ini, Anda dapat mengunduh dan menjalankan pemutakhiran driver langsung. Jika Anda tidak dapat melakukan pemutakhiran driver di tempat, Anda dapat menjalankan skrip pembantu. Skrip menentukan apakah instans Anda terpengaruh. Jika terpengaruh, dan perangkat jaringan Amazon EC2 belum dihapus, skrip menonaktifkan pemindaian Plug and Play Cleanup. Jika perangkat jaringan dihapus, skrip memperbaiki perangkat, menonaktifkan pemindaian Plug and Play Cleanup, dan memungkinkan instans Anda melakukan boot ulang dengan konektivitas jaringan diaktifkan.

Daftar Isi


- Pilih cara memperbaiki masalah
- Metode 1 - Jaringan yang ditingkatkan
- Metode 2 - Konfigurasi registri
- Jalankan skrip remediasi


Pilih cara memperbaiki masalah

Ada dua metode untuk memulihkan konektivitas jaringan dan penyimpanan ke instans yang terpengaruh oleh masalah ini. Pilih salah satu dari metode berikut:

| Metode | Prasyarat | Ringkasan Prosedur |
| --- | --- | --- |
| Metode 1 - Jaringan yang ditingkatkan | Jaringan yang ditingkatkan hanya tersedia di cloud privat virtual (VPC) yang memerluka n tipe instans C3. Jika server saat ini tidak menggunak an tipe instans C3, Anda harus mengubahnya untuk sementara. | Anda mengubah tipe instans server menjadi instans C3. Jaringan yang ditingkatkan kemudian memungkinkan Anda untuk terhubung ke instans yang terpengaruh dan memperbaiki masalah. Setelah Anda memperbaiki masalah, Anda mengubah instans kembali ke tipe instans asli. Metode ini biasanya lebih cepat daripada Metode 2 dan lebih kecil kemungkinannya dalam kesalahan pengguna. Anda akan dikenai biaya tambahan selama instans C3 berjalan. |
| Metode 2 - Konfigurasi registri | Kemampuan untuk membuat atau mengakses server kedua. Kemampuan untuk mengubah pengaturan Registri. | Anda melepaskan volume root dari instans yang terpengaruh, melampirk annya ke instans yang berbeda, menghubungkan, dan membuat perubahan di Registri. Anda akan dikenai biaya tambahan selama server tambahan berjalan. Metode ini lebih lambat daripada Metode 1, tetapi metode ini berhasil dalam situasi di mana Metode 1 gagal menyelesa ikan masalah. |

Metode 1 - Jaringan yang ditingkatkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Cari instans yang terpengaruh. Pilih instans dan pilih status Instans, lalu pilih Hentikan instans.

> **Warning**
>
> Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

4. Setelah instans dihentikan, buat cadangan. Pilih instans dan pilih Tindakan, lalu image dan templat, lalu pilih Buat image.
5. Ubah tipe instans menjadi tipe instans C3 apa pun.
6. Mulai instans.
7. Sambungkan ke instance menggunakan Remote Desktop dan kemudian unduh paket AWS PV Drivers Upgrade ke instance.
8. Ekstrak isi folder dan jalankan.`AWSPVDriverSetup.msi`

   Setelah menjalankan MSI, instans secara otomatis melakukan boot ulang kemudian memutakhirkan driver. Instans tidak akan tersedia hingga 15 menit.
9. Setelah peningkatan selesai dan instans lulus pada pemeriksaan kondisi di konsol Amazon EC2, hubungkan ke instans menggunakan Desktop Jarak Jauh dan pastikan driver baru terinstal. Di Pengelola Perangkat, di bawah Kontroler Penyimpanan, temukan Adaptor Host Penyimpanan AWS PV. Pastikan versi driver sama dengan versi terbaru yang terdaftar pada tabel Riwayat Versi Driver. Untuk informasi selengkapnya, lihat AWS Riwayat paket driver PV.
10. Hentikan instans dan ubah kembali ke tipe instans aslinya.
11. Mulai instans dan lanjutkan penggunaan normal.

Metode 2 - Konfigurasi registri

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Cari instans yang terpengaruh. Pilih instans, pilih status Instans, lalu pilih Hentikan instans.

> **Warning**
>
> Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

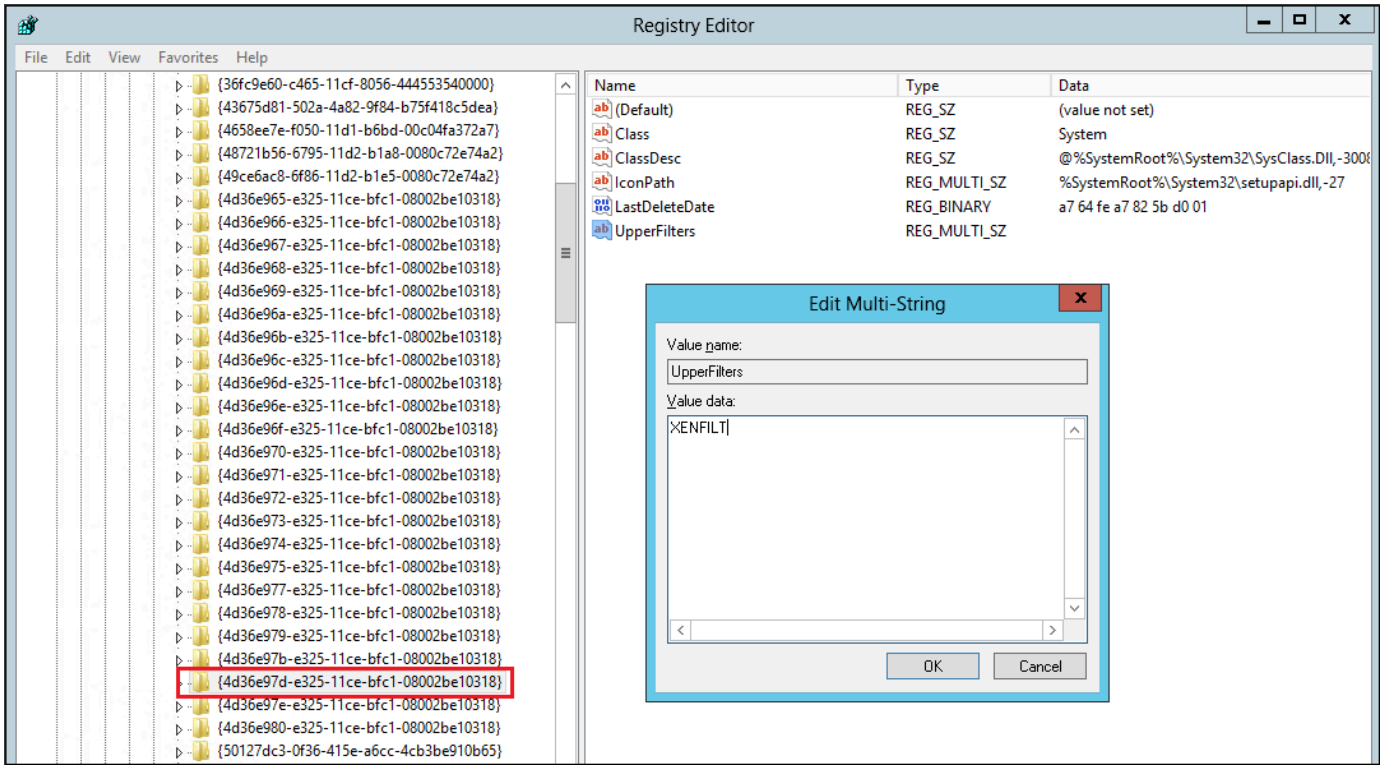
4. Pilih Luncurkan instans dan buat instans Windows Server 2008 atau Windows Server 2012 sementara di Zona Ketersediaan yang sama dengan instans yang terpengaruh. Jangan membuat instans Windows Server 2012 R2.

> **Important**
>
> Jika Anda tidak membuat instans di Zona Ketersediaan yang sama dengan instans yang terpengaruh, Anda tidak akan dapat melampirkan volume root dari instans yang terpengaruh ke instans baru.

5. Pada panel navigasi, pilih Volume.
6. Cari volume root dari instans yang terdampak. Lepaskan volume dan Lampirkan volume ke instans sementara yang Anda buat sebelumnya. Lampirkan dengan nama perangkat default (xvdf).
7. Gunakan Remote Desktop untuk terhubung ke instans sementara, dan kemudian gunakan utilitas Disk Management agar volume tersedia untuk digunakan.
8. Pada instans sementara, buka kotak dialog Jalankan, ketik **`regedit`**, dan tekan Enter.
9. Di panel navigasi Registry Editor, pilih HKEY_ Local_Machine, dan kemudian dari menu File pilih Load Hive.
10. Di kotak dialog Muat Hive, arahkan ke Volume yang Terpengaruh\Windows\System32\config \System dan ketik nama sementara di kotak dialog Nama Kunci. Misalnya, enter OldSys .
11. Di panel navigasi Editor Registri, cari kunci berikut:

    HKEY_LOCAL_MACHINE\\ ControlSet 001\ Kontrol\ Kelas***your_temporary_key_name***\ 4d36e97d-e325-11ce-bfc1-08002be10318

    HKEY_LOCAL_MACHINE\\ ControlSet 001\ Kontrol\ Kelas***your_temporary_key_name***\ 4d36e96a-e325-11ce-bfc1-08002be10318
12. Untuk setiap tombol, buka UpperFilters, masukkan nilai XENFILT, lalu pilih OK.




13. Temukan kunci berikut:

    HKEY_LOCAL_MACHINE\\ ControlSet 001\ Layanan\ X ***your_temporary_key_name*** ENBUS\ Parameter

14. Buat string baru (REG_SZ) dengan nama ActiveDevice dan nilai berikut:

    PCI\VEN_5853&DEV_0001&SUBSYS_00015853&REV_01

15. Temukan kunci berikut:

    HKEY_LOCAL_MACHINE\\ 001\ Layanan***your_temporary_key_name***\ XENBUS ControlSet

16. Ubah Count dari 0 menjadi 1.

17. Temukan dan hapus kunci berikut:

    HKEY_LOCAL_MACHINE\\ ControlSet 001\ Layanan***your_temporary_key_name***\ xenvbd\ StartOverride

    HKEY_LOCAL_MACHINE\\ ControlSet 001\ Layanan***your_temporary_key_name***\ xenfilt\ StartOverride

18. Di panel navigasi Editor Registri, pilih kunci sementara yang Anda buat saat pertama kali membuka Editor Registri.

19. Dari File pilihan, pilih Pembongkaran Hive.
20. Di Disk Management Utility, pilih drive yang Anda pasang sebelumnya, buka menu konteks (klik kanan), dan pilih Offline.
21. Di konsol Amazon EC2, lepaskan volume yang terpengaruh dari instans sementara dan pasang kembali ke instans Windows Server 2012 R2 Anda dengan nama perangkat/. dev/sda1 Anda harus menentukan nama perangkat ini untuk menetapkan volume sebagai volume root.
22. Mulai instans.
23. Sambungkan ke instance menggunakan Remote Desktop dan kemudian unduh paket AWS PV Drivers Upgrade ke instance.
24. Ekstrak isi folder dan jalankan.`AWSPVDriverSetup.msi`

    Setelah menjalankan MSI, instans secara otomatis melakukan boot ulang kemudian memutakhirkan driver. Instans tidak akan tersedia hingga 15 menit.
25. Setelah peningkatan selesai dan instans lulus pada pemeriksaan kondisi di konsol Amazon EC2, hubungkan ke instans menggunakan Desktop Jarak Jauh dan pastikan driver baru terinstal. Di Pengelola Perangkat, di bawah Kontroler Penyimpanan, temukan Adaptor Host Penyimpanan AWS PV. Pastikan versi driver sama dengan versi terbaru yang terdaftar pada tabel Riwayat Versi Driver. Untuk informasi selengkapnya, lihat AWS Riwayat paket driver PV.
26. Hapus atau hentikan instans sementara yang Anda buat dalam prosedur ini.

### Jalankan skrip remediasi

Jika Anda tidak dapat melakukan pemutakhiran driver langsung atau bermigrasi ke instans yang lebih baru, Anda dapat menjalankan skrip perbaikan untuk memperbaiki masalah yang disebabkan oleh tugas Pembersihan Pasang dan Pakai.

Untuk menjalankan skrip remediasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans yang ingin Anda jalankan skrip remediasinya. Pilih Status instans, lalu pilih Mulai instans.

> **Warning**
>
> Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

4. Setelah instans dihentikan, buat cadangan. Setelah instans, pilih Tindakan, lalu Gambar dan templat, lalu pilih Buat gambar.
5. Pilih Status instans, lalu pilih Mulai instans.
6. Hubungkan ke instance dengan menggunakan Remote Desktop dan kemudian unduh RemediateDriverIssue.zip folder ke instance.
7. Ekstrak isi folder tersebut.
8. Jalankan skrip remediasi sesuai dengan instruksi dalam Readme.txt file. File terletak di folder tempat Anda mengekstrak RemediateDriverIssue.zip.

### Pemindahan TCP

> **Important**
>
> Masalah ini tidak berlaku untuk instans yang menjalankan AWS driver jaringan PV atau Intel.

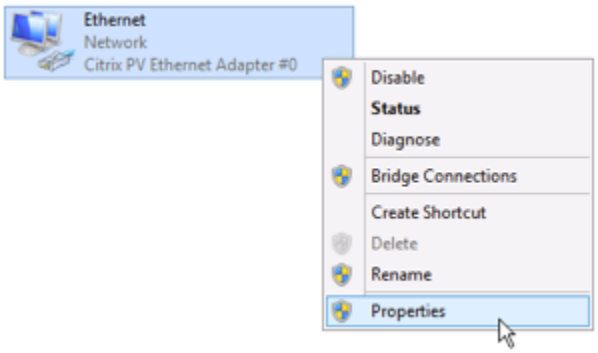
Secara default, pemindahan TCP diaktifkan untuk driver Citrix PV di AMI Windows. Jika Anda mengalami kesalahan tingkat pengangkutan atau kesalahan transmisi paket (seperti yang terlihat di Windows Performance Monitor)—misalnya, saat Anda menjalankan beban kerja SQL tertentu—Anda mungkin perlu menonaktifkan fitur ini.

> **Warning**
>
> Menonaktifkan pemindahan TCP dapat mengurangi performa jaringan instans Anda.

#### Untuk menonaktifkan pemindahan TCP untuk Windows Server 2012 dan 2008

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.

2. Jika Anda menggunakan Windows Server 2012, tekan Ctrl + Esc untuk mengakses layar Mulai, lalu pilih Panel Kontrol. Jika Anda menggunakan Windows Server 2008, pilih Mulai dan pilih Panel Kontrol.
3. Pilih Jaringan dan Internet, lalu Jaringan dan Pusat Berbagi.
4. Pilih Ubah pengaturan adaptor.
5. Buka menu konteks (klik kanan) untuk Citrix PV Ethernet Adapter #0 dan pilih Properties.




6. Di kotak dialog Properti Koneksi Area Lokal, pilih Konfigurasi untuk membuka kotak dialog Properti #0 Adaptor Ethernet Citrix PV.
7. Pada tab Advanced, nonaktifkan setiap properti, kecuali untuk Nilai TCP/UDP Checksum yang Benar. Untuk menonaktifkan properti, pilih dari Properti dan pilih Dinonaktifkan dari Nilai.
8. Pilih OKE.
9. Jalankan perintah berikut dari jendela Command Prompt.

```
netsh int ip set global taskoffload=disabled
netsh int tcp set global chimney=disabled
netsh int tcp set global rss=disabled
netsh int tcp set global netdma=disabled
```

10. Boot ulang instans.

### Sinkronisasi waktu

Sebelum rilis AMI Windows 2013.02.13, agen tamu Citrix Xen dapat salah mengatur waktu sistem. Ini dapat menyebabkan sewa DHCP Anda kedaluwarsa. Jika Anda mengalami masalah saat menghubungkan ke instans Anda, Anda mungkin perlu memperbarui agennya.

Untuk menentukan apakah Anda memiliki agen tamu Citrix Xen yang diperbarui, periksa apakah file `C:\Program Files\Citrix\XenGuestAgent.exe` dari bulan Maret 2013. Jika tanggal pada

file ini lebih awal dari itu, perbarui layanan agen tamu Citrix Xen. Untuk informasi selengkapnya, lihat Mutakhirkan layanan agen tamu Citrix Xen Anda.

Beban kerja yang memanfaatkan lebih dari 20.000 disk IOPS mengalami degradasi karena kemacetan CPU

Anda dapat terpengaruh oleh masalah ini jika Anda menggunakan instans Windows yang menjalankan driver AWS PV yang memanfaatkan lebih dari 20.000 IOPS, dan Anda mengalami kode periksa bug `0x9E: USER_MODE_HEALTH_MONITOR`.

Disk membaca dan menulis (iOS) di driver AWS PV terjadi dalam dua fase: persiapan IO dan penyelesaian IO. Secara default, tahap persiapan berjalan pada core arbiter tunggal. Tahap penyelesaian berjalan pada inti `0`. Jumlah komputasi yang diperlukan untuk memproses IO berbeda-beda berdasarkan ukuran dan properti lainnya. Beberapa iOS menggunakan komputasi lebih banyak dalam tahap persiapan, dan lainnya dalam tahap penyelesaian. Ketika sebuah instans menggerakkan lebih dari 20.000 IOPS, tahap persiapan atau penyelesaian dapat mengakibatkan hambatan, di mana CPU tempat instans tersebut berjalan ada pada kapasitas 100%. Apakah fase persiapan atau penyelesaian menjadi hambatan tergantung pada properti iOS yang digunakan oleh aplikasi.

Dimulai AWS dengan driver PV 8.4.0, beban fase persiapan dan fase penyelesaian dapat didistribusikan ke beberapa inti, menghilangkan hambatan. Setiap aplikasi menggunakan properti IO yang berbeda. Oleh karena itu, menerapkan salah satu konfigurasi berikut dapat meningkatkan, menurunkan, atau tidak mempengaruhi performa aplikasi Anda. Setelah Anda menerapkan salah satu konfigurasi ini, pantau aplikasi untuk memverifikasi bahwa aplikasi memenuhi performa yang Anda inginkan.

1. Prasyarat

   Sebelum Anda memulai prosedur pemecahan masalah ini, verifikasi prasyarat berikut:

   - Instans Anda menggunakan driver AWS PV versi 8.4.0 atau yang lebih baru. Untuk memutakhirkan, lihat Tingkatkan driver PV pada instans Windows EC2.
   - Anda memiliki akses RDP ke instans. Untuk langkah-langkah agar ter-connect ke instans Windows menggunakan RDP, lihat Sambungkan ke instance Windows Anda menggunakan klien RDP.
   - Anda memiliki akses administrator pada instans.

2. Mengamati beban CPU pada instans Anda

   Anda dapat menggunakan Windows Task Manager untuk melihat beban pada setiap CPU untuk menentukan potensi hambatan pada IO disk.

   1. Verifikasi bahwa aplikasi Anda menjalankan dan menangani lalu lintas mirip dengan beban kerja produksi Anda.
   2. Hubungkan ke instans Anda menggunakan RDP.
   3. Pilih menu Mulai pada instans Anda.
   4. Masukkan `Task Manager` di menu Mulai untuk membuka Task Manager.
   5. Jika Task Manager menampilkan Tampilan Ringkasan, pilih Detail lebih lanjut untuk menampilkan tampilan rinci.
   6. Pilih tab Performa.
   7. Pilih CPU di panel kiri.
   8. Buka menu konteks (klik kanan) untuk grafik di panel utama dan pilih U bah grafik menjadi > Proses or logis untuk menampilkan setiap inti individu.
   9. Tergantung pada berapa banyak inti pada instans, Anda mungkin melihat baris yang menampilkan beban CPU dari waktu ke waktu, atau Anda mungkin hanya melihat angka.
      - Jika Anda melihat grafik yang menampilkan beban dari waktu ke waktu, cari CPU yang hampir seluruhnya terarsir.
      - Jika Anda melihat angka pada setiap inti, cari inti yang secara konsisten menunjukkan angka 95% atau lebih besar.
   10. Perhatikan apakah inti `0` atau inti yang berbeda mengalami beban berat.

3. Pilih konfigurasi mana yang akan diterapkan

| Nama konfigurasi | Kapan harus menerapkan konfigurasi ini | Catatan |
| --- | --- | --- |
| Default configuration | Beban kerja menggerakkan kurang dari 20.000 IOPS, atau konfigurasi lain tidak meningkatkan performa atau stabilitas. | Untuk konfigurasi ini, IO terjadi pada beberapa inti, yang dapat menguntungkan beban kerja yang lebih kecil dengan meningkatkan cache |

| Nama konfigurasi | Kapan harus menerapkan konfigurasi ini | Catatan |
| --- | --- | --- |
| | | lokalitas dan mengurangi konteks beralih. |
| Allow driver to choose whether to distribute completion | Beban kerja menggerakkan lebih dari 20.000 IOPS dan beban sedang atau tinggi diamati pada inti `0`. | Konfigurasi ini direkomendasikan untuk semua instans Xen yang menggunakan PV 8.4.0 atau yang lebih baru dan memanfaatkan lebih dari 20.000 IOPS, terlepas dari masalah yang dihadapi. |
| Distribute both preparation and completion | Beban kerja menggerakkan lebih dari 20.000 IOPS, dan menyebabkan driver dapat memilih distribusi tanpa meningkatkan performa, atau inti selain `0` mengalami beban tinggi. | Konfigurasi ini memungkinkan distribusi persiapan IO dan penyelesaian IO. |

> **Note**
>
> Kami menyarankan agar Anda tidak mendistribusikan persiapan IO tanpa juga mendistribusikan penyelesaian IO (mengatur `DpcRedirection` tanpa mengatur `NotifierDistributed`) karena tahap penyelesaian peka terhadap kelebihan beban oleh tahap persiapan ketika tahap persiapan berjalan secara paralel.

Nilai-nilai kunci registri

- NotifierDistributed

  Nilai berupa `0` atau tidak ada — Tahap penyelesaian akan berjalan pada inti `0`.

  Nilai `1` — Driver memilih untuk menjalankan tahap penyelesaian atau `0` atau satu inti tambahan per disk terlampir.

Nilai 2 — Driver menjalankan tahap penyelesaian pada satu inti tambahan per disk terpasang.

- DpcRedirection

  Nilai berupa `0` atau tidak ada — Tahap persiapan akan berjalan pada inti tunggal arbiter.

  Nilai 1 — Tahap persiapan didistribusikan di banyak inti.

Konfigurasi default

Terapkan konfigurasi default dengan AWS versi driver PV sebelum 8.4.0, atau jika penurunan kinerja atau stabilitas diamati setelah menerapkan salah satu konfigurasi lain di bagian ini.

1. Hubungkan ke instans Anda menggunakan RDP.
2. Buka PowerShell command prompt baru sebagai administrator.
3. Jalankan perintah berikut untuk menghapus kunci registri `NotifierDistributed` dan `DpcRedirection`.

   ```
   Remove-ItemProperty -Path HKLM:\System\CurrentControlSet\Services\xenvbd
   \Parameters -Name NotifierDistributed
   ```

   ```
   Remove-ItemProperty -Path HKLM:\System\CurrentControlSet\Services\xenvbd
   \Parameters -Name DpcRedirection
   ```

4. Booting ulang instans Anda.

Izinkan driver untuk memilih apakah akan mendistribusikan penyelesaian

Set `NotiferDistributed` el kunci registri untuk memungkinkan driver penyimpanan PV memilih apakah akan mendistribusikan penyelesaian IO.

1. Hubungkan ke instans Anda menggunakan RDP.
2. Buka PowerShell command prompt baru sebagai administrator.
3. Jalankan perintah berikut untuk mengatur kunci registri `NotiferDistributed`.

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Value 0x00000001 -Name NotifierDistributed
```

4. Booting ulang instans Anda.

Distribusikan persiapan dan penyelesaian

Atur kunci registri `NotifierDistributed` dan `DpcRedirection` untuk selalu mendistribusikan tahap persiapan dan penyelesaian.

1. Hubungkan ke instans Anda menggunakan RDP.
2. Buka PowerShell command prompt baru sebagai administrator.
3. Jalankan perintah berikut untuk mengatur kunci registri `NotifierDistributed` dan `DpcRedirection`.

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Value 0x00000002 -Name NotifierDistributed
```

```
Set-ItemProperty -Type DWORD -Path HKLM:\System\CurrentControlSet\Services\xenvbd
\Parameters -Value 0x00000001 -Name DpcRedirection
```

4. Booting ulang instans Anda.

## NVMe driver

Volume Amazon EBS dan volume penyimpanan instans diekspos sebagai perangkat blok NVMe pada Nitro-based instans. Untuk sepenuhnya menggunakan kinerja dan kemampuan fitur Amazon EBS untuk volume yang diekspos sebagai perangkat blok NVMe, instans harus menginstal driver AWS NVMe. Semua AMI AWS Windows generasi saat ini dilengkapi dengan driver AWS NVMe yang diinstal secara default.

Untuk informasi selengkapnya tentang EBS dan NVMe, lihat Amazon EBS dan NVMe di Panduan Pengguna Amazon EBS. Untuk informasi selengkapnya tentang penyimpanan instans SSD dan NVMe, lihat Volume penyimpanan instans SSD untuk instans EC2.

### Instans Linux

AMI berikut menyertakan driver NVMe yang diperlukan:

- Amazon Linux 2
- Amazon Linux AMI 2018.03
- Ubuntu 14.04 atau yang lebih baru dengan kernel `linux-aws`

> **Note**
>
> AWS Graviton-based jenis instance memerlukan Ubuntu 18.04 atau yang lebih baru dengan kernel `linux-aws`

- Red Hat Enterprise Linux 7.4 atau setelahnya
- SUSE Linux Enterprise Server 12 SP2 atau setelahnya
- CentOS 7.4.1708 atau setelahnya
- FreeBSD 11.1 atau yang lebih baru
- Debian GNU/Linux 9 atau yang lebih baru

#### Untuk mengonfirmasi bahwa instans Anda memiliki driver NVMe

Anda dapat mengonfirmasi bahwa instans Anda memiliki driver NVMe menggunakan perintah berikut.

- Amazon Linux, RHEL, CentOS, dan SUSE Linux Server

```
$ modinfo nvme
```

  Jika instans memiliki driver NVMe, perintah mengembalikan informasi tentang driver.

- Amazon Linux 2 dan Ubuntu

```
$ ls /sys/module/ | grep nvme
```

  Jika instans memiliki driver NVMe, perintah akan mengembalikan driver yang diinstal.

Untuk memperbarui driver NVMe

Jika instans Anda memiliki driver NVMe, Anda dapat memperbarui driver ke versi terbaru dengan menggunakan prosedur berikut.

1. Terhubung ke instans Anda.
2. Perbarui cache paket Anda untuk mendapatkan pembaruan paket yang diperlukan sebagai berikut.

   - Untuk Amazon Linux 2, Amazon Linux, CentOS, dan Red Hat Enterprise Linux:

     ```
     [ec2-user ~]$ sudo yum update -y
     ```

   - Untuk Ubuntu dan Debian:

     ```
     [ec2-user ~]$ sudo apt-get update -y
     ```

3. Ubuntu 16.04 dan yang lebih baru menyertakan `linux-aws` paket, yang berisi driver NVMe dan ENA yang diperlukan oleh Nitro-based instance. Mutakhirkan paket `linux-aws` untuk menerima versi terbaru sebagai berikut.

   ```
   [ec2-user ~]$ sudo apt-get install --only-upgrade -y linux-aws
   ```

   Untuk Ubuntu 14.04, Anda dapat menginstal paket `linux-aws` sebagai berikut:

   ```
   [ec2-user ~]$ sudo apt-get install linux-aws
   ```

4. Lakukan boot ulang instans untuk memuat versi kernel terbaru.

   ```
   sudo reboot
   ```

5. Hubungkan kembali ke instans Anda setelah boot ulang.

Instans Windows

PowerShell

Jika Anda tidak meluncurkan instans dari salah satu AMI AWS Windows terbaru yang disediakan oleh Amazon, gunakan prosedur berikut untuk menginstal driver AWS NVMe saat ini pada instans

Anda. Reboot diperlukan untuk instalasi ini. Entah skrip instalasi akan mem-boot ulang instans Anda atau Anda harus mem-boot ulang sebagai langkah terakhir.

Prasyarat

- PowerShell versi 3.0 atau yang lebih baru diinstal.
- Perintah yang ditunjukkan pada bagian ini harus berjalan dalam versi 64-bit PowerShell. Jangan gunakan x86 versi PowerShell. Itu adalah versi 32-bit dari shell, dan tidak didukung untuk perintah ini.

Untuk mengunduh dan menginstal yang terbaru AWS Driver NVMe

1. Kami menyarankan Anda untuk membuat AMI sebagai cadangan sebagai berikut, jika Anda perlu mengembalikan perubahan Anda.
   a. Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Sebelum Anda menghentikan sebuah instans, pastikan bahwa Anda telah menyalin data apa pun yang Anda perlukan dari volume penyimpanan instans ke penyimpanan persisten, seperti Amazon EBS atau Amazon S3.
   b. Di panel navigasi, pilih Contoh.
   c. Pilih instans yang memerlukan pemutakhiran driver, dan pilih Status instans, Hentikan instans.
   d. Setelah instans dihentikan, pilih instans, pilih Tindakan, lalu Gambar dan templat, lalu pilih Buat gambar.
   e. Pilih Status instans, Mulai instans.
2. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
3. Unduh driver ke instance Anda menggunakan salah satu opsi berikut:
   - Browser - Unduh paket driver terbaru ke instance dan ekstrak arsip zip.
   - PowerShell— Jalankan perintah berikut:

```
Invoke-WebRequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/
NVMe/Latest/AWSNVMe.zip -outfile $env:USERPROFILE\nvme_driver.zip
Expand-Archive $env:userprofile\nvme_driver.zip -DestinationPath
 $env:userprofile\nvme_driver
```

Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:

```
[Net.ServicePointManager]::SecurityProtocol =
 [Net.SecurityProtocolType]::Tls12
```

4. Instal driver ke instance Anda dengan menjalankan `install.ps1` PowerShell skrip dari `nvme_driver` direktori (`.\install.ps1`). Jika Anda mendapatkan kesalahan, pastikan Anda menggunakan PowerShell 3.0 atau yang lebih baru.

   a. (Opsional) Dimulai dengan versi AWS NVMe`1.5.0`, reservasi persisten Small Computer System Interface (SCSI) didukung untuk Windows Server 2016 dan yang lebih baru. Fitur ini menambahkan dukungan untuk Windows Server Failover Clustering dengan penyimpanan Amazon EBS bersama. Secara default, fitur ini tidak diaktifkan selama instalasi.

   Anda dapat mengaktifkan fitur saat menjalankan skrip `install.ps1` untuk menginstal driver dengan menentukan parameter `EnableSCSIPersistentReservations` dengan nilai `$true`.

   ```
   PS C:\> .\install.ps1 -EnableSCSIPersistentReservations $true
   ```

   Anda dapat mengaktifkan fitur saat menjalankan skrip `install.ps1` untuk menginstal driver dengan menentukan parameter `EnableSCSIPersistentReservations` dengan nilai `$false`.

   ```
   PS C:\> .\install.ps1 -EnableSCSIPersistentReservations $false
   ```

   b. Dimulai dengan AWS NVMe`1.5.0`, `install.ps1` skrip selalu menginstal `ebsnvme-id` alat dengan driver.

   (Opsional) Untuk versi `1.4.0`, `1.4.1`, dan `1.4.2`, skrip `install.ps1` memungkinkan Anda untuk menentukan apakah alat `ebsnvme-id` harus diinstal dengan driver.

   i. Untuk menginstal alat `ebsnvme-id`, tentukan `InstallEBSNVMeIdTool 'Yes'`.

   ii. Jika Anda tidak ingin menginstal alat, tentukan `InstallEBSNVMeIdTool 'No'`.

Jika Anda tidak menentukan `InstallEBSNVMeIdTool`, dan alat sudah ada di `C:\ProgramData\Amazon\Tools`, paket akan memutakhirkan alat secara default. Jika alat tidak ada, `install.ps1` tidak akan memutakhirkan alat secara default.

Jika Anda tidak ingin menginstal alat sebagai bagian dari paket, dan ingin menginstalnya nanti, Anda dapat menemukan versi terbaru atau alat dalam paket driver. Atau, Anda dapat mengunduh versi `1.0.0` dari Amazon S3:

Unduh alat `ebsnvme-id`.

5. Jika penginstal tidak melakukan boot ulang instans Anda, lakukan boot ulang instans tersebut.

Distributor

Anda dapat menggunakan Distributor, kemampuan AWS Systems Manager, untuk menginstal paket driver NVMe satu kali atau dengan pembaruan terjadwal.

Untuk menginstal yang terbaru AWS Driver NVMe

1. Untuk petunjuk cara menginstal paket driver NVMe menggunakan Distributor, lihat prosedur di Menginstal atau memperbarui paket di Panduan Pengguna Amazon EC2 Systems Manager.
2. Untuk Jenis Instalasi, pilih Uninstall dan instal ulang.
3. Untuk Nama, pilih AwsnvMe.
4. (Opsional) Untuk Argum en Tambahan, Anda dapat menyesuaikan instalasi dengan menentukan nilai. Nilai harus diformat menggunakan sintaks JSON yang valid. Untuk contoh cara meneruskan argumen tambahan untuk `aws configure` paket, lihat referensi plugin dokumen perintah.

   a. Dimulai dengan AWS NVMe`1.5.0`, driver mendukung reservasi persisten SCSI untuk Windows Server 2016 dan yang lebih baru. Secara default, fitur ini tidak diaktifkan selama instalasi.

      - Untuk mengaktifkan fitur ini, tentukan`{"SSM_EnableSCSIPersistentReservations": "true"}`.
      - Jika Anda tidak ingin mengaktifkan fitur ini, tentukan`{"SSM_EnableSCSIPersistentReservations": "false"}`.

b. Dimulai dengan AWS NVMe`1.5.0`, `install.ps1` skrip akan selalu menginstal `ebsnvme-id` alat.

   (Opsional) Untuk versi `1.4.0`, `1.4.1`, dan `1.4.2`, skrip `install.ps1` memungkinkan Anda untuk menentukan apakah alat ebsnvme-id harus diinstal dengan driver.

   - Untuk menginstal alat ebsnvme-id, tentukan. `{"SSM_InstallEBSNVMeIdTool": "Yes"}`
   - Jika Anda tidak ingin menginstal alat, tentukan `{"SSM_InstallEBSNVMeIdTool": "No"}`.

     Jika tidak `SSM_InstallEBSNVMeIdTool` ditentukan untuk Argumen Tambahan, dan alat sudah ada di`C:\ProgramData\Amazon\Tools`, paket akan memutakhirkan alat secara default. Jika alat tidak ada, paket tidak akan memutakhirkan alat secara default.

     Jika Anda tidak ingin menginstal alat sebagai bagian dari paket, dan ingin menginstalnya nanti, Anda dapat menemukan versi terbaru alat dalam paket driver. Atau, Anda dapat mengunduh versi `1.0.0` dari Amazon S3:

     Unduh alat `ebsnvme-id`.

5. Jika penginstal tidak melakukan boot ulang instans Anda, lakukan boot ulang instans tersebut.

## Konfigurasikan reservasi persisten SCSI untuk instans Windows

Setelah versi driver AWS NVMe `1.5.0` atau yang lebih baru diinstal, Anda dapat mengaktifkan atau menonaktifkan reservasi persisten SCSI menggunakan registri Windows untuk Windows Server 2016 dan yang lebih baru. Anda harus melakukan boot ulang instans agar perubahan registri ini diterapkan.

Anda dapat mengaktifkan reservasi persisten SCSI dengan perintah berikut yang mengatur `EnableSCSIPersistentReservations` ke nilai `1`.

```
PS C:\> $registryPath = "HKLM:\SYSTEM\CurrentControlSet\Services\AWSNVMe\Parameters
\Device"
Set-ItemProperty -Path $registryPath -Name EnableSCSIPersistentReservations -Value 1
```

Anda dapat menonaktifkan reservasi persisten SCSI dengan perintah berikut yang mengatur `EnableSCSIPersistentReservations` ke nilai `0`.

```
PS C:\> $registryPath = "HKLM:\SYSTEM\CurrentControlSet\Services\AWSNVMe\Parameters
\Device"
Set-ItemProperty -Path $registryPath -Name EnableSCSIPersistentReservations -Value 0
```

## AWS Riwayat versi driver Windows NVMe

Tabel berikut menunjukkan driver AWS NVMe mana yang berjalan pada setiap versi Windows Server di Amazon EC2.

| Versi Windows Server | AWS Versi driver NVMe |
|---|---|
| Windows Server 2025 | versi terbaru |
| Windows Server 2022 | versi terbaru |
| Windows Server 2019 | versi terbaru |
| Windows Server 2016 | versi terbaru |
| Windows Server 2012 R2 | versi 1.5.1 dan sebelumnya |
| Windows Server 2012 | versi 1.5.1 dan sebelumnya |
| Windows Server 2008 R2 | versi 1.3.2 dan sebelumnya |
| Windows Server 2008 | versi 1.3.2 dan sebelumnya |

Tabel berikut menjelaskan versi driver AWS NVMe yang dirilis.

| Versi paket | Versi Driver | Detail | Tanggal rilis |
|---|---|---|---|
| 1.8.2 1.8.2 | 1.8.2 | • Memperbaiki pemeriksaan bug saat startup dengan meningkatkan penanganan interupsi. | Juni 15, 2026 |

| Versi paket | Versi Driver | Detail | Tanggal rilis |
| --- | --- | --- | --- |
| 1.8.1 1.8.1 | 1.8.1 | • Memperbaiki masalah kompatibilitas mundur dengan pembuatan nomor seri disk. | Februari 24, 2026 |
| 1.8.0 | 1.8.0 | • Menambahkan dukungan untuk iOCTL untuk beroperasi dengan perangkat yang menerapkan manajemen namespace dinamis. Untuk informasi selengkapnya, lihat IOCTL_SCSI_MINIPORT IOCTL di dokumentasi Microsoft.<br>• Perbaikan bug dalam perintah Get Log Page dan pembuatan nomor seri. | Januari 16, 2026 |
| 1.7.0 1.7.0 | 1.7.0 | • Menambahkan dukungan untuk NVMe Get Log Page Command.<br>• Menambahkan dukungan untuk statistik kinerja terperinci untuk volume penyimpanan instans EBS dan EC2.<br>• Perbaikan bug untuk perintah Identify Namespace. | 17 September 2025 |
| 1.6.0 1.6.0 | 1.6.0 | • Memperbarui skrip instalasi untuk menggunakan PnPutil.<br>• Diperbarui ebsnvme-id.exe untuk menggunakan NVMe IOCTL. | Oktober 25, 2024 |
| 1.5.1 1.5.1 | 1.5.0 | Memperbaiki skrip instalasi untuk membuat folder untuk `ebsnvme-id` alat jika tidak ada. | 17 November 2023 |

| Versi paket | Versi Driver | Detail | Tanggal rilis |
| --- | --- | --- | --- |
| 1.5.0 | 1.5.0 | Menambahkan dukungan untuk reservasi persisten Small Computer System Interface (SCSI) untuk instans yang menjalankan Windows Server 2016 dan setelahnya. Alat ebsnvme-id (`ebsnvme-id.exe`) sekarang diinstal secara default. | 31 Agustus 2023 |
| 1.4.2 | 1.4.2 | Memperbaiki bug di mana Driver AWS NVMe tidak mendukung volume penyimpanan instance pada instance D3. | 16 Maret 2023 |
| 1.4.1 | 1.4.1 | Laporan Namespace Preferred Write Granularity (NPGW) untuk volume EBS yang mendukung fitur NVMe opsional ini. Untuk informasi lebih lanjut, lihat bagian 8.25, “Meningkatkan Kinerja melalui Kepatuhan I/O Ukuran dan Penyelarasan,” di Spesifikasi Das ar NVMe, versi 1.4. | Mei 20, 2022 |
| 1.4.0 | 1.4.0 | • Menambahkan dukungan untuk IOCTL yang memungkinkan aplikasi berinteraksi dengan perangkat NVMe. Dukungan ini memungkinkan aplikasi untuk mendapatkan daftar `IdentifyController`, `IdentifyNamespace`, dan `NameSpace` dari perangkat NVMe. Untuk informasi selengkapnya, lihat Protocol-specific kueri dalam dokumentasi Microsoft.<br>• Versi driver 1.4.0 dan alat `ebsnvme-id` terbaru (`ebsnvme-id.exe`) digabungkan dalam satu paket. Kombinasi ini memungkinkan Anda untuk menginstal driver dan alat dari satu paket. Untuk detail selengkapnya, lihat NVMe driver.<br>• Perbaikan bug dan peningkatan keandalan. | 23 November 2021 |

| Versi paket | Versi Driver | Detail | Tanggal rilis |
|---|---|---|---|
| 1.3.2 1.3.2 | 1.3.2 | Memperbaiki masalah pengubahan volume EBS yang secara aktif memproses IO, yang dapat mengakibatkan kerusakan data. Pelanggan yang tidak memodifikasi volume EBS online (misalnya, mengubah ukuran atau mengubah tipe) tidak akan terpengaruh.<br><br>Ini adalah versi terakhir yang dapat berjalan pada Windows Server 2008 dan 2008 R2. Versi ini tersedia untuk diunduh tetapi tidak lagi didukung. Windows Server 2008 dan 2008 R2 telah mencapai akhir masa pakai, dan tidak lagi didukung oleh Microsoft. | 10 September 2019 |
| 1.3.1 | 1.3.1 | Peningkatan keandalan. | 21 Mei 2019 |
| 1.3.0 | 1.3.0 | Peningkatan pengoptimalan perangkat. | 31 Agustus 2018 |
| 1.2.0 | 1.2.0 | Peningkatan kinerja dan keandalan untuk perangkat AWS NVMe pada semua instans yang didukung, termasuk instans bare metal. | 13 Juni 2018 |
| > 1.0.0 | > 1.0.0 | AWS Driver NVMe untuk jenis instance yang didukung menjalankan Windows Server. | 12 Februari 2018 |

Berlangganan notifikasi

Amazon SNS dapat memberi Anda notifikasi saat EC2 Windows Drivers versi baru dirilis. Gunakan prosedur berikut untuk berlangganan notifikasi ini.

Untuk berlangganan notifikasi EC2 dari konsol

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di bilah navigasi, ubah Wilayah menjadi AS Timur (Virginia Utara), jika perlu. Anda harus memilih Wilayah ini karena notifikasi SNS langganan Anda ada di Wilayah ini.

3. Di panel navigasi, pilih Langganan.
4. Pilih Buat langganan.
5. Di kotak dialog Create subscription (Buat langganan), lakukan hal berikut:
   a. Untuk TopicARN, salin Amazon Resource Name (ARN) berikut:

      arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers
   b. Untuk Protokol, pilih `Email`.
   c. Untuk Titik Akhir, ketik alamat email yang bisa Anda gunakan untuk menerima notifikasi.
   d. Pilih Buat langganan.
6. Anda akan menerima email konfirmasi. Buka email dan ikuti petunjuk untuk menyelesaikan langganan Anda.

Setiap kali driver EC2 Windows baru dirilis, kami mengirimkan notifikasi ke pelanggan. Jika Anda tidak ingin lagi menerima notifikasi ini, gunakan prosedur berikut untuk berhenti berlangganan.

Untuk berhenti berlangganan dari notifikasi driver Windows Amazon EC2

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di panel navigasi, pilih Berlangganan.
3. Pilih kotak centang untuk langganan lalu pilih Tindakan, Hapus langganan. Ketika diminta konfirmasi, pilih Hapus.

Untuk berlangganan notifikasi EC2 menggunakan AWS CLI

Untuk berlangganan notifikasi EC2 dengan AWS CLI, gunakan perintah berikut.

```
aws sns subscribe --topic-arn arn:aws:sns:us-east-1:801119661308:ec2-windows-drivers --
protocol email --notification-endpoint YourUserName@YourDomainName.ext
```

Untuk berlangganan notifikasi EC2 menggunakan AWS Tools for Windows PowerShell

Untuk berlangganan notifikasi EC2 dengan AWS Tools for Windows PowerShell, gunakan perintah berikut.

```
Connect-SNSNotification -TopicArn 'arn:aws:sns:us-east-1:801119661308:ec2-windows-
drivers' -Protocol email -Region us-east-1 -Endpoint 'YourUserName@YourDomainName.ext'
```

# Driver perangkat Windows lainnya

Perangkat AWS VMClock adalah perangkat jam virtual yang ada pada Nitro-based instance. Perangkat muncul di Pengelola Perangkat di bawah Perangkat sistem sebagai`AWS VMClock Device`. AWS menyediakan paket driver null (no-function) untuk Windows yang memungkinkan sistem operasi mengenali perangkat.

## Instans-instans yang didukung

Perangkat AWS VMClock tersedia pada jenis Nitro-based instans. Untuk daftar jenis Nitro-based instance, lihat Nitro-based instance.

## Sistem operasi yang didukung

Paket driver mendukung versi Windows Server berikut:

- Windows Server 2016
- Windows Server 2019
- Windows Server 2022
- Windows Server 2025

## Instalasi driver

Jika perangkat AWS VMClock menunjukkan ikon peringatan di Device Manager, yang menunjukkan bahwa tidak ada driver yang diinstal, Anda dapat menginstal paket driver null secara manual.

Untuk menginstal AWS Pengemudi VMClock

1. Unduh dan ekstrak paket driver dari Amazon S3.

```
Invoke-WebRequest https://s3.amazonaws.com/ec2-windows-drivers-downloads/
AWSVMClock/Latest/AWSVMClock.zip -OutFile $env:USERPROFILE\AWSVMClock.zip
Expand-Archive $env:USERPROFILE\AWSVMClock.zip -DestinationPath $env:USERPROFILE
\AWSVMClock
```

2. Instal driver menggunakan`pnputil`.

```
pnputil /add-driver $env:USERPROFILE\AWSVMClock\aws_vmclock.inf /install
```

3. Verifikasi status perangkat di Device Manager. Anda tidak perlu me-reboot instance.

# Konfigurasikan instans Windows Amazon EC2 Anda

Setelah meluncurkan instance Windows, Anda dapat masuk sebagai administrator untuk melakukan konfigurasi tambahan untuk fitur Windows dan pengaturan sistem. Utilitas pemecahan masalah Windows EC2dapat membantu Anda memecahkan masalah pada instans Anda.

Anda dapat mengonfigurasi agen peluncuran Windows dan Windows-specific fitur lainnya sebagai berikut.

Agen peluncuran Windows

Setiap AMI AWS Windows (dan banyak AMI lain yang tersedia di AWS Marketplace) menyertakan agen peluncuran Windows yang telah dikonfigurasi sebelumnya dengan pengaturan default. Agen peluncuran melakukan tugas selama startup instans dan dijalankan jika instance dihentikan dan kemudian dimulai, atau dimulai ulang.

Peluncuran Cepat EC2 untuk Windows

Setiap instans Windows Amazon EC2 harus melalui langkah-langkah peluncuran sistem operasi Windows standar (OS), yang mencakup beberapa boot ulang, dan seringkali membutuhkan waktu 15 menit atau lebih lama untuk menyelesaikannya. AMI Amazon EC2 Windows Server yang mengaktifkan fitur Peluncuran Cepat EC2 menyelesaikan beberapa langkah tersebut dan melakukan reboot terlebih dahulu untuk mengurangi waktu yang dibutuhkan untuk meluncurkan instance.

## Windows-specific pengaturan sistem

Daftar berikut mencakup beberapa pengaturan sistem yang hanya berlaku untuk sistem operasi Windows:

Ubah kata sandi Administrator Windows

Saat Anda terhubung ke instans Windows, Anda harus menentukan akun pengguna dan kata sandi yang memiliki izin untuk mengakses instans. Pertama kali Anda terhubung ke instance, Anda harus menggunakan akun Administrator dan memberikan kata sandi default. Saat Anda terhubung ke sebuah instans untuk pertama kalinya, kami menyarankan Anda untuk mengubah kata sandi Administrator dari nilai default-nya.

Tambahkan komponen Sistem Windows

Sistem operasi Windows Server mencakup banyak komponen opsional. Termasuk semua komponen opsional di setiap AWS Windows Server AMI tidak praktis. Sebagai gantinya, kami menyediakan snapshot EBS media instalasi yang memiliki file yang diperlukan untuk mengonfigurasi atau menginstal komponen pada instans Windows Anda.

Instal WSL di Windows

Windows Subsystem for Linux (WSL) adalah unduhan gratis yang dapat Anda instal di instans Windows Anda. Dengan menginstal WSL, Anda dapat menjalankan alat baris perintah Linux native secara langsung di instans Windows Anda dan menggunakan alat Linux untuk membuat skrip, di samping desktop Windows tradisional Anda. Anda dapat dengan mudah bertukar antara Linux dan Windows pada satu instans Windows, yang mungkin berguna bagi Anda dalam lingkungan pengembangan.

## AWS driver perangkat untuk instans Windows

Anda dapat memperbarui driver AWS perangkat untuk instans Windows Anda. Untuk informasi selengkapnya, lihat the section called “Kelola driver perangkat”.

Tabel berikut merangkum driver yang didukung untuk Nitro-based instance berdasarkan versi Windows.

| Versi | Pengemudi penyimpanan | Driver jaringan yang ditingkatkan |
|---|---|---|
| Windows Server 2025 | AWS NVMe versi terbaru | ENA versi terbaru |
| Windows Server 2022 | AWS NVMe versi terbaru | ENA versi terbaru |
| Windows Server 2019 | AWS NVMe versi terbaru | ENA versi terbaru |
| Windows Server 2016 | AWS NVMe versi terbaru | ENA versi terbaru |
| Windows Server 2012 R2 | AWS NVMe versi 1.5.1 | ENA versi 2.6.0 |
| Windows Server 2008 R2 | AWS NVMe versi 1.3.2 | ENA versi 2.2.3 |

Tabel berikut merangkum driver yang didukung untuk Xen-based instance berdasarkan versi Windows.

| Versi | Pengemudi penyimpanan | Driver jaringan yang ditingkat kan |
| --- | --- | --- |
| Windows Server 2022 | AWS PV versi terbaru | • ENA versi terbaru [1]<br>• Intel VF 2<br>• AWS PV versi terbaru [3] |
| Windows Server 2019 | AWS PV versi terbaru | • ENA versi terbaru [1]<br>• Intel VF 2<br>• AWS PV versi terbaru [3] |
| Windows Server 2016 | AWS PV versi terbaru | • ENA versi terbaru [1]<br>• Intel VF 2<br>• AWS PV versi terbaru [3] |
| Windows Server 2012 R2 | AWS Versi PV 8.4.3 | • ENA versi 2.6.0 1<br>• Intel VF 2<br>• AWS Versi PV 8.4.3 3 |
| Windows Server 2008 R2 | AWS Versi PV 8.3.5 | • ENA versi 2.2.3 1<br>• Intel VF 2<br>• AWS Versi PV 8.3.5 3 |

[1] Misalnya tipe G3, H1, I3,, P3`m4.16xlarge`, P3dn, dan R4.

[2] Misalnya tipe C3, C4, D2, I2, M4 (tidak termasuk`m4.16xlarge`), dan R3.

[3] Misalnya tipe C1, M1, M2, M3, T1, T2, X1, dan X1e.

# Agen peluncuran Windows di instans Windows Amazon EC2

Setiap AMI AWS Windows menyertakan agen peluncuran Windows yang telah dikonfigurasi sebelumnya dengan pengaturan default. Agen peluncuran melakukan tugas selama startup instans dan dijalankan jika instance dihentikan dan kemudian dimulai, atau dimulai ulang. Untuk informasi tentang agen tertentu, lihat halaman detail dalam daftar berikut.

Untuk informasi selengkap AWS nya tentang AMI Windows, lihat referensi AWS Windows AMI.

- Gunakan agen EC2launch v2 untuk melakukan tugas selama peluncuran instance EC2 Windows
- Gunakan agen EC2launch v1 untuk melakukan tugas selama peluncuran instance EC2 Windows
- Gunakan layanan EC2config untuk melakukan tugas selama peluncuran instance sistem operasi Windows lama EC2

Daftar isi


- Bandingkan agen peluncuran Amazon EC2
- Konfigurasikan Sufiks DNS untuk agen peluncuran EC2 Windows
- Berlangganan pemberitahuan agen peluncuran EC2 Windows
- Administrasi Layanan Windows untuk agen EC2launch v2 dan EC2config


## Bandingkan agen peluncuran Amazon EC2

Tabel berikut menunjukkan perbedaan fungsional utama antara EC2Config, EC2Lauch v1, dan EC2Launch v2.

| Fitur | EC2Config | EC2Launch v1 | EC2Launch v2 |
|---|---|---|---|
| Jalankan sebagai | Layanan Windows | PowerShell Skrip | Layanan Windows |
| Mendukung | Hanya OS warisan | Versi Windows Server:<br>• 2016<br>• 2019 (LTSC dan SAC) | Versi Windows Server:<br>• 2016<br>• 2019 (LTSC dan SAC)<br>• 2022 |

| Fitur | EC2Config | EC2Launch v1 | EC2Launch v2 |
| --- | --- | --- | --- |
| | | | • 2025 |
| File konfigurasi | XML | JSON | JSON/YAML |
| Tetapkan nama pengguna Administrator | Tidak | Tidak | Ya |
| Data pengguna terkompresi | Tidak | Tidak | Ya |
| Data pengguna lokal dibuat di AMI | Tidak | Tidak | Ya, dapat dikonfigurasi |
| Konfigurasi tugas dalam data pengguna | Tidak | Tidak | Ya |
| Wallpaper yang dapat dikonfigurasi | Tidak | Tidak | Ya |
| Sesuaikan urutan jalannya tugas | Tidak | Tidak | Ya |
| Tugas yang dapat dikonfigurasi | 15 | 9 | 20 saat peluncuran |
| Mendukung Windows Event Viewer | Ya | Tidak | Ya |
| Jumlah tipe peristiwa Penampil Peristiwa | 2 | 0 | 30 |

> **Note**
>
> Dokumentasi EC2config disediakan untuk referensi historis saja. Versi sistem operasi yang dijalankannya tidak lagi didukung oleh Microsoft. Kami sangat menyarankan Anda meningkatkan ke layanan peluncuran terbaru.

## Konfigurasikan Sufiks DNS untuk agen peluncuran EC2 Windows

Dengan agen peluncuran Amazon EC2, Anda dapat mengonfigurasi daftar sufiks DNS yang digunakan instans Windows untuk resolusi nama domain. Agen peluncuran mengganti pengaturan Windows standar dalam kunci `System\CurrentControlSet\Services\Tcpip\Parameters\SearchList` registri dengan menambahkan nilai-nilai berikut ke daftar pencarian akhiran DNS:

- Domain instance
- Sufiks yang dihasilkan dari devolusi domain instance
- Domain NV
- Domain yang ditentukan oleh setiap kartu antarmuka jaringan

Semua agen peluncuran mendukung konfigurasi akhiran DNS. Untuk informasi selengkapnya, lihat versi agen peluncuran spesifik Anda:

- Untuk informasi tentang `setDnsSuffix` tugas dan cara mengkonfigurasi sufiks DNS di EC2launch v2, lihat. set DnsSuffix
- Untuk informasi tentang pengaturan daftar akhiran DNS dan cara mengaktifkan atau menonaktifkan devolusi untuk EC2launch v1, lihat. Konfigurasikan agen EC2launch v1 pada instans Windows Anda
- Untuk informasi tentang pengaturan daftar akhiran DNS dan cara mengaktifkan atau menonaktifkan devolusi untuk EC2config, lihat. File pengaturan EC2Config

### Devolusi nama domain

Devolusi nama domain adalah perilaku Direktori Aktif yang memungkinkan komputer di domain anak mengakses sumber daya di domain induk tanpa menggunakan nama domain yang sepenuhnya memenuhi syarat. Secara default, devolusi nama domain berlanjut sampai hanya ada dua node yang tersisa dalam perkembangan nama domain.

Agen peluncuran melakukan devolusi pada nama domain jika instance terhubung ke domain, dan menambahkan hasil ke daftar pencarian akhiran DNS yang disimpan dalam kunci registri **`System\CurrentControlSet\Services\Tcpip\Parameters\SearchList`**. Agen menggunakan pengaturan dari kunci registri berikut, untuk menentukan perilaku devolusi.

- **`System\CurrentControlSet\Services\Tcpip\Parameters\UseDomainNameDevolution`**
  - Saat tidak disetel, menonaktifkan devolusi
  - Saat disetel ke`1`, mengaktifkan devolusi (default)
  - Saat disetel ke`0`, menonaktifkan devolusi
- **`System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel`**
  - Jika tidak disetel, gunakan level 2 (default)
  - Saat disetel ke 3 atau lebih besar, gunakan nilai untuk mengatur level

Saat Anda menonaktifkan devolusi atau mengubah pengaturan devolusi ke tingkat yang lebih tinggi, gaya kunci `System\CurrentControlSet\Services\Tcpip\Parameters\SearchList` registri berisi sufiks yang ditambahkan sebelumnya. Mereka tidak dihapus secara otomatis. Anda dapat memperbarui daftar secara manual, atau Anda dapat menghapus daftar dan membiarkan agen Anda menjalankan proses untuk menyiapkan daftar baru.

> **Note**
>
> Untuk menghapus daftar akhiran DNS dari registri, Anda dapat menjalankan perintah berikut.
>
> ```
> PS C:\> Invoke-CimMethod -ClassName Win32_NetworkAdapterConfiguration -
> MethodName "SetDNSSuffixSearchOrder" -Arguments @{ DNSDomainSuffixSearchOrder =
>  $null } | Out-Null
> ```

Contoh devolusi

Contoh berikut menunjukkan perkembangan nama domain melalui proses devolusi.

`corp.example.com`

- Berlanjut ke `example.com`

`locale.region.corp.example.com`

1. Berlanjut ke `region.corp.example.com`
2. Berlanjut ke `corp.example.com`
3. Berlanjut ke `example.com`

`locale.region.corp.example.com`dengan pengaturan `DomainNameDevolutionLevel=3`

1. Berlanjut ke `region.corp.example.com`
2. Berlanjut ke`corp.example.com`. Perkembangan berhenti di sini, karena pengaturan level.

## Berlangganan pemberitahuan agen peluncuran EC2 Windows

Amazon SNS dapat memberi tahu Anda ketika versi baru agen peluncuran EC2 dirilis. Gunakan prosedur berikut untuk berlangganan notifikasi ini.

Berlangganan notifikasi EC2config

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di bilah navigasi, ubah Wilayah menjadi AS Timur (Virginia Utara), jika perlu. Anda harus memilih Wilayah ini karena notifikasi SNS tempat Anda berlangganan dibuat di Wilayah ini.
3. Di panel navigasi, pilih Langganan.
4. Pilih Buat langganan.
5. Di kotak dialog Create subscription (Buat langganan), lakukan hal berikut:
   a. Untuk  Topik ARN, gunakan Nama Sumber Daya Amazon (ARN) berikut yang cocok dengan agen yang ingin Anda terima notifikasi:
      - EC2Launch v2:

        ```
        arn:aws:sns:us-east-1:309726204594:amazon-ec2launch-v2
        ```

      - EC2launch atau EC2config:

        ```
        arn:aws:sns:us-east-1:801119661308:ec2-windows-ec2config
        ```

   b. Untuk Protokol, pilih `Email`.

c. Untuk Endpoint, masukkan alamat email tempat Anda ingin menerima notifikasi.

d. Pilih Buat langganan.

6. Anda akan menerima email yang meminta Anda untuk mengonfirmasi langganan Anda. Buka email dan ikuti petunjuk untuk menyelesaikan langganan Anda.

Setiap kali versi baru agen peluncuran dirilis, kami mengirim pemberitahuan ke pelanggan. Jika Anda tidak ingin lagi menerima notifikasi ini, gunakan prosedur berikut untuk berhenti berlangganan.

Berhenti berlangganan dari pemberitahuan agen peluncuran

1. Buka konsol Amazon SNS.
2. Di panel navigasi, pilih Langganan.
3. Pilih langganan lalu pilih Tindakan, Hapus langganan. Ketika diminta untuk mengonfirmasi, pilih Hapus.

## Administrasi Layanan Windows untuk agen EC2launch v2 dan EC2config

Jika Anda telah masuk ke instans Anda sebagai pengguna dengan hak administratif, Anda dapat mengelola agen peluncuran EC2launch v2 dan EC2config seperti halnya layanan Windows lainnya. EC2launch v1 adalah sekumpulan PowerShell skrip yang dikelola melalui tugas terjadwal secara default. Bagian ini mencakup administrasi layanan untuk EC2launch v2 dan EC2config.

Untuk menerapkan pengaturan yang diperbarui ke instans Anda, Anda dapat menghentikan dan memulai ulang agen EC2Launch v2 atau agen peluncuran layanan EC2Config dari antarmuka Microsoft Management Console (MMC) untuk Layanan. Demikian pula, ketika Anda menginstal versi baru agen peluncuran, Anda harus menghentikan agen terlebih dahulu, lalu memulai ulang ketika instalasi selesai.

> **Note**
>
> Anda harus membuka antarmuka Layanan MMC sebagai administrator untuk memilih tindakan ini. Untuk melakukan ini, Anda dapat memilih J alankan sebagai administrator dari menu konteks. Atau, untuk membuka antarmuka menggunakan keyboard Anda, ikuti langkah-langkah ini:
>
> 1. Menggunakan Tab tombol atau tombol panah, pilih item menu Layanan dari menu Alat Administratif.

2. Gunakan kombinasi keyboard berikut untuk membuka sebagai administrator: `Ctrl + Shift +Enter`.

Prosedur berikut mencantumkan langkah-langkah untuk menghentikan dan memulai agen peluncuran pada instans Anda.

Hentikan agen peluncuran

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pilih Alat Administ ratif dari menu Start Windows.
3. Buka konsol Layanan sebagai administrator, seperti yang dijelaskan di awal bagian ini.
4. Dalam daftar layanan, pilih agen yang berjalan pada instans Anda (EC2Launch atau EC2Config), lalu pilih Stop dari menu Action. Atau, Anda dapat menggunakan menu konteks untuk menghentikan agen.

Mulai ulang agen peluncuran

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pilih Alat Administ ratif dari menu Start Windows.
3. Buka konsol Layanan sebagai administrator, seperti yang dijelaskan di awal bagian ini.
4. Dalam daftar layanan, pilih agen yang berjalan pada instans Anda (EC2Launch atau EC2Config), lalu pilih Mulai atau Mulai Ulang dari menu T ind akan. Atau, Anda dapat menggunakan menu konteks untuk memulai ulang agen.

Jika Anda tidak perlu memperbarui pengaturan konfigurasi, membuat AMI Anda sendiri, atau menggunakan AWS Systems Manager, Anda dapat menghapus atau menghapus agen peluncuran.

Hapus

Menghapus layanan akan menghapus subkunci registrinya.

Copot pemasangan

Menghapus instalasi layanan akan menghapus file, subkunci registri, dan pintasan apa pun ke layanan tersebut.

Hapus agen peluncuran

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Mulai jendela Prompt Perintah Windows.
3. Jalankan salah satu perintah berikut untuk menghapus agen peluncuran.
   - Jalankan perintah berikut untuk menghapus EC2launch atau EC2launch v2:

     ```
     sc delete ec2launch
     ```

   - Jalankan perintah berikut untuk menghapus layanan EC2config:

     ```
     sc delete ec2config
     ```

Copot pemasangan agen peluncuran

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Pilih Sistem Windows, lalu Panel Kontrol dari menu Start Windows.
3. Pilih Program dan Fitur untuk membuka daftar program yang diinstal pada instans Anda.
4. Pilih agen peluncuran Anda dari daftar (Amazon EC2launch atau EC2ConfigService), lalu pilih Uninstall dari menu File. Atau, Anda dapat menggunakan menu konteks.

> **Note**
>
> Anda dapat melihat versi agen peluncuran apa yang diinstal di kolom Versi.

## Gunakan agen EC2launch v2 untuk melakukan tugas selama peluncuran instance EC2 Windows

Semua instans Amazon EC2 yang didukung yang diluncurkan dari AMI AWS Windows Server 2022 dan Windows Server 2025 menyertakan agen peluncuran EC2launch v2 (`EC2Launch.exe`) secara default. Kami juga menyediakan AMI Windows Server 2016 dan 2019 dengan EC2launch v2 yang diinstal sebagai agen peluncuran default. AMI ini disediakan selain AMI Windows Server 2016 dan 2019 yang menyertakan EC2launch v1. Anda dapat mencari AMI Windows yang menyertakan EC2Launch v2 secara default dengan memasukkan prefiks berikut dalam pencarian Anda dari halaman AMI di konsol Amazon EC2: `EC2LaunchV2-Windows_Server-*`.

Untuk membandingkan fitur versi agen peluncuran, lihatBandingkan agen peluncuran Amazon EC2.

EC2Launch v2 melakukan tugas-tugas selama startup instans dan berjalan jika sebuah instans dihentikan dan kemudian dimulai, atau dimulai ulang. EC2Launch v2 juga dapat melakukan tugas sesuai permintaan. Beberapa dari tugas ini diaktifkan secara otomatis, sementara yang lainnya harus diaktifkan secara manual. Layanan EC2Launch v2 mendukung semua fitur EC2Config dan EC2Launch.

Layanan ini menggunakan file konfigurasi untuk mengontrol operasinya. Anda dapat memperbarui file konfigurasi dengan menggunakan alat grafis atau dengan mengeditnya secara langsung sebagai file .yml tunggal (`agent-config.yml`). Untuk informasi selengkapnya tentang lokasi file, lihatStruktur direktori EC2Launch v2.

EC2Launch v2 menerbitkan log peristiwa Windows untuk membantu Anda memecahkan masalah kesalahan dan mengatur pemicu. Untuk informasi selengkapnya, lihat Log peristiwa Windows.

Versi OS yang didukung

Agen EC2Launch v2 mendukung versi sistem operasi (OS) Windows Server berikut:

- Windows Server 2025
- Windows Server 2022
- Windows Server 2019 (Sal Long-Term uran dan Sal Semi-Annual uran Servis)
- Windows Server 2016

Tugas yang dijalankan secara default

Agen EC2Launch v2 menjalankan tugas berikut satu kali hanya secara default selama peluncuran instans awal. Tugas diatur sesuai dengan urutan di mana mereka berjalan dalam tahap peluncuran mereka.

`Bootpanggung`

- memperpanjang RootPartition

`PreReadypanggung`

- activateWindows

- set DnsSuffix
- set AdminAccount
- setWallpaper

`PostReady`panggung

- startSsm

### Konsep EC2Launch v2

Konsep berikut berguna untuk dipahami saat mempertimbangkan EC2Launch v2.

konfigurasi agen

`agent-config` adalah file yang terletak di folder konfigurasi untuk EC2Launch v2. Ini termasuk konfigurasi untuk boot, jaringanPreReady, dan ta PostReady hapan. File ini digunakan untuk menentukan konfigurasi instans untuk tugas-tugas yang harus dijalankan saat AMI di-boot untuk pertama kali atau untuk waktu-waktu berikutnya.

Secara default, instalasi EC2Launch v2 menginstal file `agent-config` yang mencakup konfigurasi yang direkomendasikan yang digunakan dalam AMI Amazon Windows standar. Anda dapat memperbarui file konfigurasi untuk mengubah pengalaman boot default untuk AMI Anda yang ditentukan EC2Launch v2. Untuk informasi selengkapnya tentang lokasi file, lihatStruktur direktori EC2Launch v2.

Frekuensi

Frekuensi tugas menentukan kapan tugas harus dijalankan, tergantung pada konteks boot. Sebagian besar tugas hanya memiliki satu frekuensi yang diizinkan. Anda dapat menentukan frekuensi untuk tugas `executeScript`.

Anda akan melihat frekuensi berikut di Konfigurasi EC2Launch v2.

- Once - Tugas dijalankan sekali, saat AMI telah boot untuk pertama kali (selesai Sysprep).
- Selalu — Tugas berjalan setiap kali agen peluncuran berjalan. Agen peluncuran berjalan saat:
  - sebuah instans dimulai atau dimulai ulang
  - layanan EC2Launch berjalan
  - `EC2Launch.exe run` diinvokasi

Panggung

Tahap adalah pengelompokan logis dari tugas yang dijalankan agen EC2Launch v2. Beberapa tugas hanya dapat dijalankan dalam tahap tertentu. Yang lain dapat berjalan dalam beberapa tahap. Saat menggunakan `agent-config.yml`, Anda harus menentukan daftar tahapan, dan daftar tugas untuk dijalankan dalam setiap tahap.

Layanan berjalan tahapan dalam urutan sebagai berikut:

Tahap 1: Boot

Tahap 2: Jaringan

Tahap 3: PreReady

Windows sudah siap

Setelah PreReady tahap selesai, layanan mengirimkan `Windows is ready` pesan ke konsol Amazon EC2.

Tahap 4: PostReady

Data pengguna berjalan selama PostReady tahap. Beberapa versi skrip berjalan sebelum PostReady tahap `agent-config.yml` file, dan beberapa berjalan setelahnya, sebagai berikut:

Sebelum `agent-config.yml`

- Data pengguna YAML versi 1.1
- Data pengguna XML

Setelah `agent-config.yml`

- Data pengguna YAMB versi 1.0 (versi warisan untuk kompatibilitas mundur)

Untuk contoh tahapan dan tugas, lihat Contoh: `agent-config.yml`.

Saat Anda menggunakan data pengguna, Anda harus menentukan daftar tugas agar agen peluncuran dijalankan. Panggung tersirat. Untuk contoh tugas, lihat Contoh: data pengguna.

EC2launch v2 menjalankan daftar tugas dalam urutan yang Anda tentukan dalam `agent-config.yml` dan dalam data pengguna. Tahapan berjalan secara berurutan. Tahap selanjutnya dimulai setelah tahap sebelumnya selesai. Tugas juga berjalan secara berurutan.

Tugas

Anda dapat menginvokasi tugas untuk melakukan tindakan pada sebuah instans. Anda dapat mengonfigurasi tugas dalam file `agent-config.yml` atau melalui data pengguna. Untuk daftar tugas yang tersedia untuk EC2Launch v2, lihat Tugas EC2Launch v2. Untuk skema konfigurasi tugas dan detailnya, lihat Konfigurasi EC2Launch v2.

Data pengguna

Data pengguna adalah data yang dapat dikonfigurasi saat Anda meluncurkan sebuah instans. Anda dapat memperbarui data pengguna agar secara dinamis mengubah bagaimana AMI kustom atau AMI mulai cepat dikonfigurasi. EC2Launch v2 mendukung panjang input data pengguna 60 kB. Data pengguna hanya mencakup UserData tahap, dan karena itu berjalan setelah `agent-config` file. Anda dapat memasukkan data pengguna ketika Anda meluncurkan sebuah instans menggunakan wizard peluncuran instans, atau Anda dapat memodifikasi data pengguna dari konsol EC2. Untuk informasi lebih lanjut tentang bekerja dengan data pengguna, lihat Cara Amazon EC2 menangani data pengguna untuk instans Windows.

Ikhtisar tugas EC2launch v2

EC2Launch v2 dapat melakukan tugas berikut di setiap boot:

- Siapkan wallpaper baru dan yang disesuaikan secara opsional yang menyajikan informasi tentang instans.
- Setel atribut untuk akun administrator yang dibuat di mesin lokal.
- Tambahkan sufiks DNS ke daftar sufiks pencarian. Hanya sufiks yang belum ada yang ditambahkan ke daftar.
- Atur huruf drive untuk volume tambahan dan perluas untuk menggunakan ruang yang tersedia.
- Tulis file dari konfigurasi ke disk.
- Jalankan skrip yang ditentukan dalam file konfigurasi EC2launch v2 atau dari. `user-data` Skrip dari `user-data` dapat berupa teks biasa atau di-zip dan disediakan sebagai format base64.
- Jalankan program dengan argumen yang diberikan.
- Tetapkan nama komputer.
- Kirim informasi instans ke konsol Amazon EC2.
- Kirim sidik jari sertifikat RDP ke konsol Amazon EC2.
- Secara dinamis, perluas partisi sistem operasi untuk menyertakan ruang yang tidak dipartisi.

- Jalankan data pengguna. Untuk informasi selengkapnya tentang menentukan data pengguna, lihat Konfigurasi EC2Launch v2.
- Setel rute statis non-persisten untuk menjangkau layanan metadata dan server. AWS KMS
- Atur partisi non-boot ke `mbr` ataugpt.
- Mulai layanan Systems Manager setelah Sysprep.
- Optimalkan pengaturan ENA.
- Aktifkan OpenSSH untuk versi Windows yang lebih baru.
- Aktifkan Jumbo Frame.
- Atur Sysprep untuk menjalankan EC2Launch v2.
- Publikasikan log peristiwa Windows.

### Struktur direktori EC2Launch v2

EC2Launch v2 harus diinstal di direktori berikut:

- Biner layana: `%ProgramFiles%\Amazon\EC2Launch`
- Data layanan (pengaturan, file log, dan file statu): `%ProgramData%\Amazon\EC2Launch`

> **Note**
>
> Secara default, Windows menyembunyikan file dan folder dalam `C:\ProgramData`. Untuk melihat direktori dan file EC2Launch v2, Anda harus memasukkan jalur di Windows Explorer atau ubah properti folder untuk menampilkan file dan folder tersembunyi.

Direktori `%ProgramFiles%\Amazon\EC2Launch` berisi binari dan pustaka pendukung. Ini mencakup subdirektori berikut:

- `settings`
  - `EC2LaunchSettingsUI.exe`— antarmuka pengguna untuk memodifikasi `agent-config.yml` file
  - `YamlDotNet.dll`- DLL untuk mendukung beberapa operasi di antarmuka pengguna
- `tools`
  - `ebsnvme-id.exe`— alat untuk memeriksa metadata volume EBS pada instance

  - `AWSAcpiSpcrReader.exe`— alat untuk menentukan port COM yang benar untuk digunakan
  - `EC2LaunchEventMessage.dll`- DLL untuk mendukung logging acara Windows untuk EC2launch
- `service`
  - `EC2LaunchService.exe`- Layanan Windows yang dapat dieksekusi yang diluncurkan ketika agen peluncuran berjalan sebagai layanan
- `EC2AgentTelemetry.dll`- DLL untuk mendukung telemetri agen EC2
- `EC2Launch.exe`— EC2launch utama dapat dieksekusi
- `EC2LaunchAgentAttribution.txt`— atribusi untuk kode yang digunakan dalam Peluncuran EC2

Direktori `%ProgramData%\Amazon\EC2Launch` berisi subdirektori berikut. Semua data yang dihasilkan oleh layanan, termasuk log, konfigurasi, dan status, disimpan di direktori ini.

- `config`— Konfigurasi

  File konfigurasi layanan disimpan dalam direktori ini sebagai `agent-config.yml`. File ini dapat diperbarui untuk mengubah, menambah, atau menghapus tugas default yang dijalankan oleh layanan. Izin untuk membuat file di direktori ini dibatasi untuk akun administrator untuk mencegah eskalasi hak istimewa.
- `log`— Log instans

  Log untuk layanan (`agent.log`) dan telemetri (`telemetry.log`) disimpan di direktori ini. Ketika `agent.log` mencapai ukuran 1 MB, itu secara otomatis diputar dan file cadangan dibuat dengan format stempel waktu (misalnya,`agent-2026-03-02T18-56-39.188.log`). Hanya satu file log cadangan yang dipertahankan pada satu waktu.
- `state`— Data status layanan

  Status yang digunakan layanan untuk menentukan tugas mana yang harus dijalankan disimpan di sini. Ada sebuah file `.run-once` yang menunjukkan apakah layanan telah dijalankan setelah Sysprep (jadi tugas dengan frekuensi sekali akan dilewati pada proses berikutnya). Subdirektori ini mencakup `state.json` dan `previous-state.json` untuk melacak status setiap tugas.
- `sysprep`— Sysprep

  Direktori ini berisi file yang digunakan untuk menentukan operasi mana yang akan dilakukan oleh Sysprep saat membuat AMI Windows kustom yang dapat digunakan kembali.

- `wallpaper`— Kertas Dinding

  Gambar wallpaper ini disimpan di direktori ini.

### Telemetri

Telemetri adalah informasi tambahan yang AWS membantu memahami kebutuhan Anda dengan Layanan AWS lebih baik, mendiagnosis masalah, dan memberikan fitur untuk meningkatkan pengalaman Anda.

EC2Launch v2 versi `2.1.592` dan setelahnya mengumpulkan telemetri, seperti metrik penggunaan dan kesalahan. Data ini dikumpulkan dari instans Amazon EC2 tempat EC2Launch v2 dijalankan. Ini termasuk semua AMI Windows yang dimiliki oleh AWS.

Tipe telemetri berikut dikumpulkan oleh EC2Launch v2:

- Informasi penggunaan — perintah agen, metode penginstalan, dan frekuensi eksekusi terjadwal.
- Kesalahan dan informasi diagnostik — kode kesalahan instalasi agen, menjalankan kode kesalahan, dan tumpukan panggilan kesalahan.

Contoh data yang dikumpulkan dari versi 2.0.592 hingga 2.1.1:

```
2025/07/18 22:38:52Z: EC2LaunchTelemetry: IsTelemetryEnabled=true
2025/07/18 22:38:52Z: EC2LaunchTelemetry: AgentOsArch=windows_amd64
2025/07/18 22:38:52Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2025/07/18 22:38:52Z: EC2LaunchTelemetry: AgentCommandErrorCode=0
2025/07/18 22:38:52Z: EC2LaunchTelemetry: AdminPasswordTypeCode=0
2025/07/18 22:38:52Z: EC2LaunchTelemetry: IpConflictDetectionCode=0
2025/07/18 22:38:52Z: EC2LaunchTelemetry: AgentErrorLocation=addroutes.go:49
```

Dimulai dengan versi 2.2.63, data telemetri Agen EC2 diformat sebagai objek JSON:

```
{"type":"EC2AgentTelemetry","agentId":"WindowsLaunchAgentV2" ... }
```

Telemetri tidak diaktifkan secara default. Anda dapat menonaktifkan kumpulan telemetri kapan saja.

#### Menonaktifkan telemetri pada sebuah instans

Untuk menonaktifkan telemetri untuk satu instans, Anda dapat mengatur variabel lingkungan sistem, atau menggunakan MSI untuk memodifikasi instalasi.

Untuk menonaktifkan telemetri dengan menyetel variabel lingkungan sistem, jalankan perintah berikut sebagai administrator.

```
setx /M EC2LAUNCH_TELEMETRY 0
```

Untuk menonaktifkan telemetri menggunakan MSI, jalankan perintah berikut setelah Anda mengunduh MSI.

```
msiexec /i ".\AmazonEC2Launch.msi" Remove="Telemetry" /q
```

Topik lainnya untuk EC2launch v2

- Instal EC2Launch v2 versi terbaru
- Konfigurasikan pengaturan EC2launch v2 untuk instans Windows
- Definisi tugas untuk tugas startup EC2launch v2
- Memecahkan masalah dengan agen EC2launch v2
- Riwayat versi EC2Launch v2

### Instal EC2Launch v2 versi terbaru

Anda dapat menggunakan salah satu metode berikut untuk menginstal agen EC2Launch v2 pada instans EC2 Anda:

- Unduh agen dari Amazon S3 dan instal dengan Windows PowerShell. Untuk mengunduh URL, lihat Unduhan EC2Launch v2 di Amazon S3.
- Instal dengan Distributor SSM.
- Instal dari komponen EC2 Image Builder saat Anda membuat gambar khusus.
- Luncurkan instans Anda dari AMI yang telah diinstal sebelumnya EC2launch v2.

> **Warning**
>
> AmazonEC2Launch.msi menghapus versi sebelumnya dari layanan peluncuran EC2, seperti EC2launch (v1) dan EC2config.

Untuk langkah penginstalan, pilih tab yang cocok dengan metode pilihan Anda.

PowerShell

Untuk menginstal versi terbaru agen EC2launch v2 dengan Windows PowerShell, ikuti langkah-langkah ini.

1. Buat direktori lokal Anda.

```
New-Item -Path "$env:USERPROFILE\Desktop\EC2Launchv2" -ItemType Directory
```

2. Tetapkan URL untuk lokasi unduhan Anda. Jalankan perintah berikut dengan URL Amazon S3 yang akan Anda gunakan. Untuk mengunduh URL, lihat Unduhan EC2Launch v2 di Amazon S3.

```
$Url = "Amazon S3 URL/AmazonEC2Launch.msi"
```

3. Gunakan perintah majemuk berikut untuk mengunduh agen dan menjalankan penginstalan

```
$DownloadFile = "$env:USERPROFILE\Desktop\EC2Launchv2\" + $(Split-Path -Path
 $Url -Leaf)
Invoke-WebRequest -Uri $Url -OutFile $DownloadFile
msiexec /i "$DownloadFile"
```

> **Note**
>
> Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:
>
> ```
> [Net.ServicePointManager]::SecurityProtocol =
>  [Net.SecurityProtocolType]::Tls12
> ```

4. Per msiexec intah menginstal EC2launch v2 di lokasi berikut pada instans Windows Server:. `%ProgramFiles%\Amazon\EC2Launch` Untuk memverifikasi bahwa penginstalan berjalan, Anda dapat memeriksa sistem file lokal pada instance Anda.

AWS Systems Manager Distributor

Untuk mengkonfigurasi pembaruan otomatis untuk EC2launch v2 dengan AWS Systems Manager Quick Setup, lihat. Instal dan perbarui secara otomatis dengan Penyiapan Cepat Distributor

Anda juga dapat melakukan penginstalan `AWSEC2Launch-Agent` paket satu kali dari AWS Systems Manager Distributor. Untuk instruksi tentang cara menginstal paket dari Systems Manager Distributor, lihat Menginstal atau memperbarui paket di Panduan Pengguna AWS Systems Manager .

EC2 Image Builder component

Anda dapat menginstal komponen `ec2launch-v2-windows` saat membuat gambar kustom dengan EC2 Image Builder. Untuk petunjuk tentang cara membuat gambar kustom dengan EC2 Image Builder, lihat Buat pipeline gambar menggunakan wizard konsol EC2 Image Builder di Panduan Pengguna EC2 Image Builder.

AMI

EC2launch v2 sudah diinstal sebelumnya secara default pada AMI untuk sistem operasi Windows Server 2022 dan di atasnya:

- Windows_Server-*version*-English-Full-Base
- Windows_Server-*version*-English-Core-Base
- Windows_Server- *version* - English-Core-EKS _Dioptimalkan
- AMI Windows *version* Server dengan semua bahasa lainnya
- AMI Windows *version* Server dengan SQL diinstal

EC2Launch v2 sudah diinstal sebelumnya di AMI Windows Server berikut. Anda dapat menemukan AMI ini dari konsol Amazon EC2, atau dengan menggunakan prefiks pencarian berikut: `EC2LaunchV2-` di AWS CLI.

- EC2LaunchV2-Windows_Server-2019-English-Core-Base
- EC2LaunchV2-Windows_Server-2019-English-Full-Base
- EC2LaunchV2-Windows_Server-2016-English-Core-Base
- EC2LaunchV2-Windows_Server-2016-English-Full-Base

### Instal dan perbarui EC2launch v2 secara otomatis dengan AWS Systems Manager Pengaturan Cepat Distributor

Dengan Pen AWS Systems Manager yiapan Cepat Distributor, Anda dapat mengatur pembaruan otomatis untuk EC2launch v2. Proses berikut menyiapkan Asosiasi Manajer Sistem pada instans Anda yang secara otomatis memperbarui agen EC2launch v2 pada frekuensi yang Anda tentukan. Asosiasi yang dibuat oleh Pengaturan Cepat Distributor dapat menyertakan instance dalam Akun AWS dan Wilayah, atau instance dalam AWS Organisasi. Untuk informasi selengkapnya tentang menyiapkan organisasi, lihat Tutorial: Membuat dan mengonfigurasi organisasi di Panduan AWS Organizations Pengguna.

Sebelum memulai, pastikan instans Anda memenuhi semua prasyarat.

#### Prasyarat

Untuk mengatur pembaruan otomatis dengan Penyiapan Cepat Distributor, instans Anda harus memenuhi prasyarat berikut.

- Anda memiliki setidaknya satu instans yang sedang berjalan yang mendukung EC2launch v2. Lihat sistem operasi yang didukung untukEC2Launch v2.
- Anda telah melakukan tugas penyiapan Manajer Sistem pada instans Anda. Untuk informasi selengkapnya, lihat Men yiapkan Manajer Sistem di Panduan AWS Systems Manager Pengguna.
- EC2launch v2 harus menjadi satu-satunya agen peluncuran yang diinstal pada instans Anda. Jika Anda menginstal lebih dari satu agen peluncuran, konfigurasi Penyiapan Cepat Distributor Anda akan gagal. Sebelum Anda mengonfigurasi EC2launch v2 dengan Penyiapan Cepat Distributor, hapus instalan agen peluncuran EC2config atau EC2launch v1, jika ada.

#### Konfigurasikan Pengaturan Cepat Distributor untuk EC2launch v2

Untuk membuat konfigurasi untuk EC2launch v2 dengan Penyiapan Cepat Distributor, gunakan pengaturan berikut saat Anda menyelesaikan langkah-langkah untuk penerapan paket Distributor:

- Paket perangkat lunak: agen Amazon EC2launch v2.
- Perbarui frekuensi: Pilih frekuensi dari daftar.
- Target: Pilih dari opsi penerapan yang tersedia.

Untuk memeriksa status konfigurasi Anda, navigasikan ke tab Konfigurasi Pengaturan Cepat Manajer Sistem di Konsol Manajemen AWS.

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Pengaturan Cepat.
3. Di tab Konfigurasi, pilih baris yang terkait dengan konfigurasi yang Anda buat. Tab Konfigurasi mencantumkan konfigurasi Anda, dan menyertakan ringkasan detail utama, seperti Wil ayah, Status Pener apan, dan status A sosiasi.

   > **Note**
   >
   > Nama asosiasi untuk setiap konfigurasi Distributor EC2launch v2 dimulai dengan awalan berikut:. `AWS-QuickSetup-Distributor-EC2Launch-Agent-`

4. Untuk melihat detail, pilih konfigurasi dan pilih Lihat detail.

Untuk informasi selengkapnya dan langkah-langkah pemecahan masalah, lihat Memecahkan Masalah Hasil Peng aturan Cepat di Panduan AWS Systems Manager Pengguna.

Unduhan EC2Launch v2 di Amazon S3

Untuk menginstal versi terbaru EC2launch v2, unduh penginstal dari lokasi berikut:

- https://s3.amazonaws.com/amazon-ec2launch-v2/windows/amd64/latest/AmazonEC2Launch.msi

Konfigurasikan opsi instalasi

Saat Anda menginstal atau memutakhirkan EC2Launch v2, Anda dapat mengonfigurasi opsi instalasi dengan dialog instalasi EC2Launch v2 atau dengan perintah msiexec di shell baris perintah.

Pertama kali penginstal EC2launch v2 berjalan pada sebuah instans, ini menginisialisasi pengaturan agen peluncuran pada instans Anda sebagai berikut:

- Ini menciptakan jalur lokal dan menulis file agen peluncuran ke sana. Hal ini terkadang disebut sebagai instalasi bersih.
- Ini menciptakan variabel `EC2LAUNCH_TELEMETRY` lingkungan jika belum ada, dan menetapkannya berdasarkan konfigurasi Anda.

Untuk detail konfigurasi, pilih tab yang cocok dengan metode konfigurasi yang akan Anda gunakan.

## Amazon EC2Launch Setup dialog

Saat Anda menginstal atau memutakhirkan EC2Launch v2, Anda dapat mengonfigurasi opsi instalasi berikut melalui dialog instalasi EC2Launch v2.

Instalasi Dasar options

Kirim Telemetri

Saat Anda menyertakan fitur ini dalam dialog penyiapan, penginstal mengatur variabel lingkungan `EC2LAUNCH_TELEMETRY` ke nilai `1`. Jika Anda menonaktifkan Kirim Telemetri, penginstal menetapkan variabel lingkungan ke nilai. `0`

Saat agen EC2launch v2 berjalan, agen ini membaca variabel lingkungan `EC2LAUNCH_TELEMETRY` untuk menentukan apakah akan mengunggah data telemetri. Jika nilainya sama dengan 1, agen menggunggah data. Jika tidak, itu tidak mengunggah.

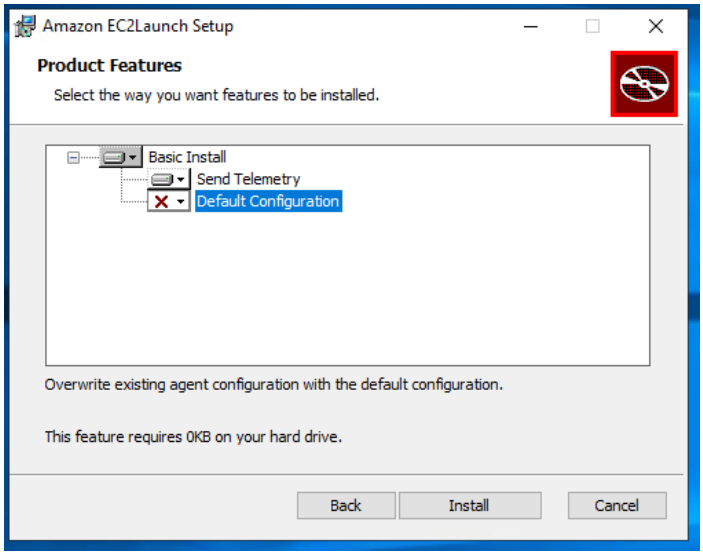
Konfigurasi default

Konfigurasi default untuk EC2launch v2 adalah menimpa agen peluncuran lokal jika sudah ada. Pertama kali Anda menjalankan instalasi pada sebuah instans, konfigurasi default melakukan instalasi bersih. Jika Anda menonaktifkan konfigurasi default pada instalasi awal, instalasi gagal.

Jika Anda menjalankan instalasi lagi pada instans, Anda dapat menonaktifkan konfigurasi default untuk melakukan pemutakhiran yang tidak menggantikan file `%ProgramData%/Amazon/EC2Launch/config/agent-config.yml`.

Contoh: Mutakhirkan EC2Launch v2 dengan telemetri

Contoh berikut menunjukkan dialog pengaturan EC2launch v2 yang dikonfigurasi untuk memutakhirkan instalasi saat ini dan mengaktifkan telemetri. Konfigurasi ini melakukan instalasi tanpa mengganti file konfigurasi agen, dan menetapkan variabel lingkungan `EC2LAUNCH_TELEMETRY` ke nilai `1`.




Command line

Saat Anda menginstal atau memutakhirkan EC2Launch v2, Anda dapat mengonfigurasi opsi instalasi berikut dengan perintah msiexec di shell baris perintah.

**`Nilai`** parameter ADDLOCAL

Dasar (wajib)

Instal agen peluncuran. Jika nilai ini tidak ada dalam `ADDLOCAL` parameter, instalasi berakhir.

Bersih

Saat Anda menyertakan nilai `Clean` dalam parameter `ADDLOCAL`, penginstal menuliskan file konfigurasi agen ke lokasi berikut: `%ProgramData%/Amazon/EC2Launch/config/agent-config.yml`. Jika file konfigurasi agen sudah ada, file tersebut akan menimpa file.

Saat Anda membiarkan nilai `Clean` keluar dari parameter `ADDLOCAL`, penginstal melakukan pemutakhiran yang tidak menggantikan file konfigurasi agen.

Telemetri

Ketika Anda memasukkan nilai `Telemetry` dalam parameter `ADDLOCAL`, pengisntal mengatur variabel lingkungan `EC2LAUNCH_TELEMETRY` ke nilai `1`.

Ketika Anda membiarkan nilai `Telemetry` keluar dari parameter `ADDLOCAL`, pengintal menetapkan variabel lingkungan ke nilai `0`.

Saat agen EC2launch v2 berjalan, agen ini membaca variabel lingkungan `EC2LAUNCH_TELEMETRY` untuk menentukan apakah akan mengunggah data telemetri. Jika nilainya sama dengan 1, agen menggunggah data. Jika tidak, itu tidak mengunggah.

Contoh: instal EC2Launch v2 dengan telemetri

```
& msiexec /i "C:\Users\Administrator\Desktop\EC2Launchv2\AmazonEC2Launch.msi"
 ADDLOCAL="Basic,Clean,Telemetry" /q
```

Verifikasi versi EC2Launch v2

Gunakan salah satu prosedur berikut untuk memverifikasi versi EC2Launch v2 yang diinstal pada instans Anda.

PowerShell

Verifikasi versi EC2launch v2 yang diinstal dengan Windows PowerShell, sebagai berikut.

1. Luncurkan sebuah instans dari AMI dan hubungkan diri Anda dengan instans tersebut.
2. Jalankan perintah berikut PowerShell untuk memverifikasi versi EC2launch v2 yang diinstal:

```
& "C:\Program Files\Amazon\EC2Launch\EC2Launch.exe" version
```

Windows Control Panel

Verifikasi versi EC2Launch v2 yang diinstal di Panel Kontrol Windows, sebagai berikut.

1. Luncurkan sebuah instans dari AMI dan hubungkan diri Anda dengan instans tersebut.
2. Buka Panel Kontrol Windows dan pilih Program dan Fitur.
3. Cari `Amazon EC2Launch` dalam daftar program yang diinstal. Nomor versinya muncul di kolom Versi.

Untuk melihat pembaruan terbaru untuk AMI AWS Windows, lihat riwayat versi AMI Windows di Refer AWS ensi AMI Windows.

Untuk EC2Launch v2 versi terbaru, lihat Riwayat versi EC2Launch v2.

Anda dapat menerima notifikasi saat layanan EC2Launch v2 versi baru dirilis. Lihat informasi yang lebih lengkap di Berlangganan pemberitahuan agen peluncuran EC2 Windows.

## Konfigurasikan pengaturan EC2launch v2 untuk instans Windows

Bagian ini berisi informasi tentang cara mengonfigurasi pengaturan untuk EC2Launch v2.

Topiknya mencakup:

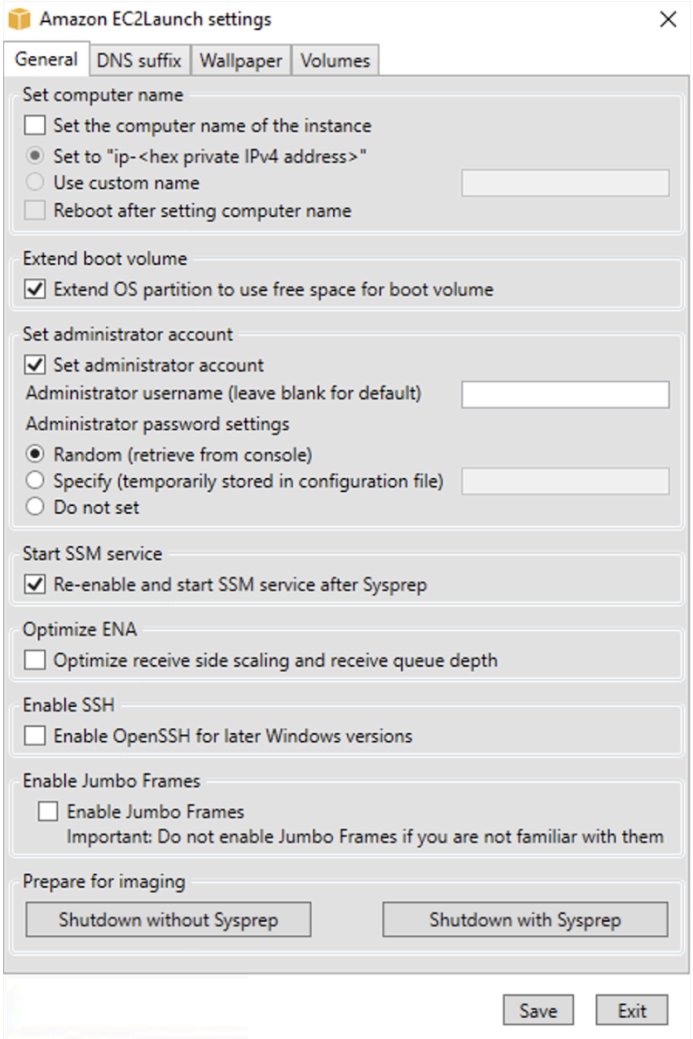
- Ubah pengaturan menggunakan kotak dialog pengaturan EC2Launch v2
- Konfigurasikan EC2Launch v2 menggunakan CLI
- Konfigurasi EC2Launch v2
- Kode keluar dan boot ulang EC2Launch v2
- EC2Launch v2 dan Sysprep

### Ubah pengaturan menggunakan kotak dialog pengaturan EC2Launch v2

Prosedur berikut menjelaskan cara menggunakan kotak dialog pengaturan EC2Launch v2 untuk mengaktifkan atau menonaktifkan pengaturan.

> **Note**
>
> Jika Anda mengonfigurasi tugas kustom secara tidak benar di file agent-config.yml. dan Anda mencoba membuka kotak dialog pengaturan Amazon EC2Launch, Anda akan menerima kesalahan. Untuk contoh skema, lihat Contoh: `agent-config.yml`.

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Dari menu Start, pilih Semua Program, lalu navigasikan ke pengaturan EC2Launch. Sebelum memilih  Shutdown with Sysprep  atau  Shutdown tanpa Sysprep, pastikan Anda menyimpan perubahan apa pun yang ingin Anda terapkan saat menjalankan shutdown.




3. Pada tab Umum dari kotak dialog pengaturan EC2Launch, Anda dapat mengaktifkan atau menonaktifkan pengaturan berikut.

   a. Atur Nama Komputer

      Jika pengaturan ini diaktifkan (dinonaktifkan secara default), maka nama host saat ini dibandingkan dengan nama host yang diinginkan di setiap boot. Jika nama host tidak cocok, maka nama host disetel ulang, dan sistem kemudian secara opsional melakukan boot ulang untuk mengambil nama host baru. Jika nama host kustom tidak ditentukan, maka akan dihasilkan menggunakan alamat IPv4 privat dengan format heksadesimal, misalnya, `ip-AC1F4E6`. Untuk mencegah perubahan nama host yang ada, jangan aktifkan pengaturan ini.

   b. Perpanjang Volume Boot

Pengaturan ini secara dinamis memperluas `Disk 0/Volume 0` untuk memasukkan ruang yang tidak dipartisi. Ini dapat berguna ketika instance di-boot dari volume root yang memiliki ukuran khusus.

c. Atur Akun Administrator

Saat diaktifkan, Anda dapat mengatur atribut nama pengguna dan kata sandi untuk akun administrator yang dibuat di mesin lokal Anda. Jika fitur ini tidak diaktifkan, akun administrator tidak dibuat di sistem setelah Sysprep. Berikan kata sandi dalam`adminPassword` hanya jika`adminPasswordtype` adalah`Specify`.

Jenis kata sandi ditentukan sebagai berikut:

i. `Random`

EC2Launch menghasilkan dan mengenkripsikan kata sandi menggunakan kunci pengguna. Sistem akan menonaktifkan pengaturan ini setelah instans dijalankan sehingga kata sandi ini akan tetap ada jika instans tersebut di-boot ulang atau dihentikan dan dimulai.

ii. `Specify`

EC2Launch menggunakan kata sandi yang Anda tentukan di `adminPassword`. Jika kata sandi tidak memenuhi persyaratan sistem, maka EC2Lauch membuat kata sandi acak sebagai gantinya. Kata sandi disimpan di `agent-config.yml` sebagai teks polos dan dihapus setelah Sysprep mengatur kata sandi administrator. EC2Launch mengenkripsi kata sandi menggunakan kunci pengguna.

iii. `Do not set`

EC2Launch menggunakan kata sandi yang Anda tentukan di file unattend.xml. Jika Anda tidak menentukan kata sandi di unattend.xml, akun administrator dinonaktifkan.

d. Mulai Layanan SSM

Ketika dipilih, layanan Systems Manajger diaktifkan untuk mulai mengikuti Sysprep. EC2Launch v2 melakukan semua tugas yang dijelaskan sebelumnya, dan SSM Agent memproses permintaan untuk kapabilitas Systems Manager, seperti Run Command dan State Manager.

Anda dapat menggunakan Run Command untuk memutakhirkan instans yang ada untuk mengunakan pada layanan EC2Launch v2 dan SSM Agent versi terbaru. Untuk informasi

selengkapnya, lihat Memperbarui Agen SSM menggunakan Run Command di Panduan AWS Pengguna Manajer Sistem.

e. Optimalkan ENA

Saat dipilih, pengaturan ENA dikonfigurasi untuk memastikan bahwa pengaturan ENA Receive Side Scaling dan Receive Queue Depth dioptimalkan AWS. Untuk informasi selengkapnya, lihat Konfigurasikan afinitas CPU penskalaan sisi Terima.

f. Aktifkan SSH

Pengaturan ini memungkinkan OpenSSH untuk versi Windows yang lebih baru untuk memungkinkan administrasi sistem jarak jauh.

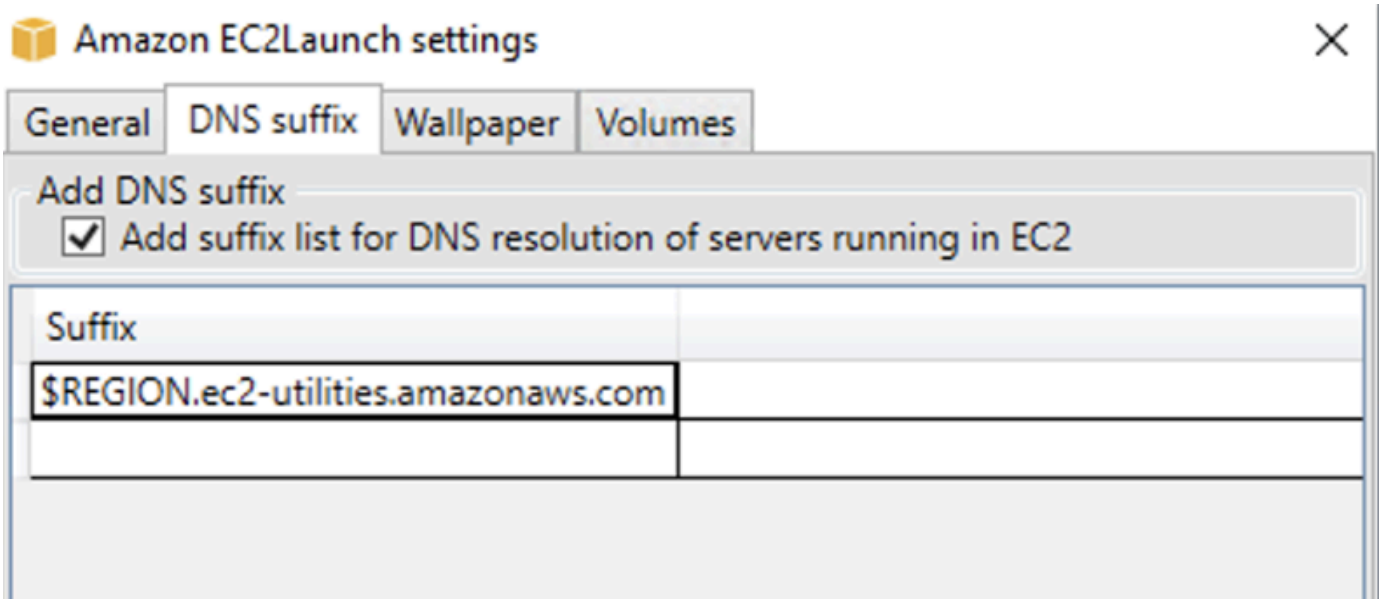
g. Aktifkan Jumbo Frame

Pilih untuk mengaktifkan Jumbo Frames. Jumbo Frames dapat memiliki efek yang tidak diinginkan pada komunikasi jaringan Anda, jadi pastikan Anda memahami bagaimana Jumbo Frames akan memengaruhi sistem Anda sebelum mengaktifkan. Untuk informasi selengkapnya tentang Jumbo Frames, lihat Frame jumbo (9001 MTU).

h. Persiapkan untuk Pencitraan

Pilih apakah Anda ingin instans EC2 Anda dimatikan dengan atau tanpa Sysprep. Saat Anda ingin menjalankan Sysprep dengan EC2Launch v2, pilih Matikan dengan Sysprep.

4. Pada tab Sufiks DNS, Anda dapat memilih apakah Anda ingin menambahkan daftar sufiks DNS untuk resolusi DNS dari server yang berjalan di EC2, tanpa memberikan nama domain yang memenuhi syarat. Sufiks DNS dapat berisi variabel `$REGION` dan `$AZ`. Hanya sufiks yang belum ada yang akan ditambahkan ke daftar.

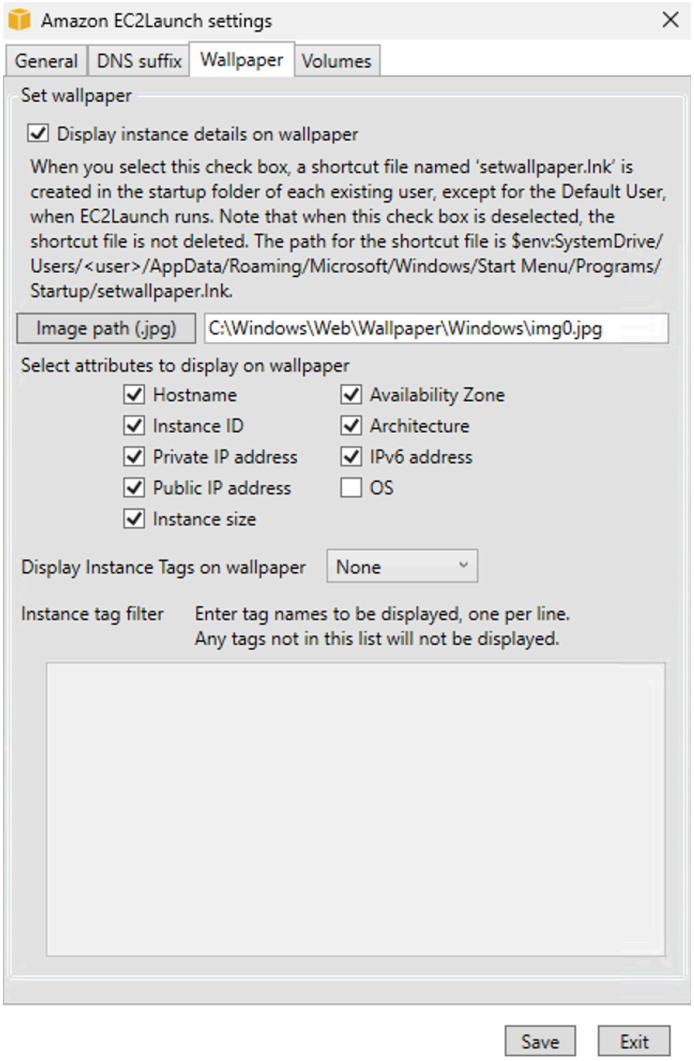



5. Pada tab Wallpaper, Anda dapat mengonfigurasi wallpaper instans Anda dengan gambar latar belakang, dan menentukan detail instans untuk wallpaper yang akan ditampilkan. Amazon EC2 menghasilkan detail setiap kali Anda masuk.

   Anda dapat mengonfigurasi wallpaper Anda dengan kontrol berikut.

   - Tampilkan detail instans pada wallpaper — Kotak centang ini mengaktifkan atau menonaktifkan tampilan detail instans pada wallpaper.
   - Jalur gambar (.jpg) - Tentukan jalur ke gambar yang akan digunakan sebagai latar belakang wallpaper.
   - Pilih atribut yang akan ditampilkan di wallpaper — Pilih kotak centang untuk detail instans yang ingin Anda tampilkan di wallpaper. Hapus kotak centang untuk detail instans yang dipilih sebelumnya yang akan Anda hapus dari wallpaper.
   - Tampilkan Tanda Instans pada wallpaper - Pilih salah satu pengaturan berikut untuk menampilkan tanda instans pada wallpaper:
     - Tidak ada - Jangan tampilkan tanda instans apa pun di wallpaper.
     - Tampilkan semua — Tampilkan semua tanda instans pada wallpaper.
     - Tampilkan difilter - Tampilkan tanda instans tertentu pada wallpaper. Saat memilih pengaturan ini, Anda dapat menambahkan tanda instans yang ingin ditampilkan di wallpaper di kotak filter tanda instans.

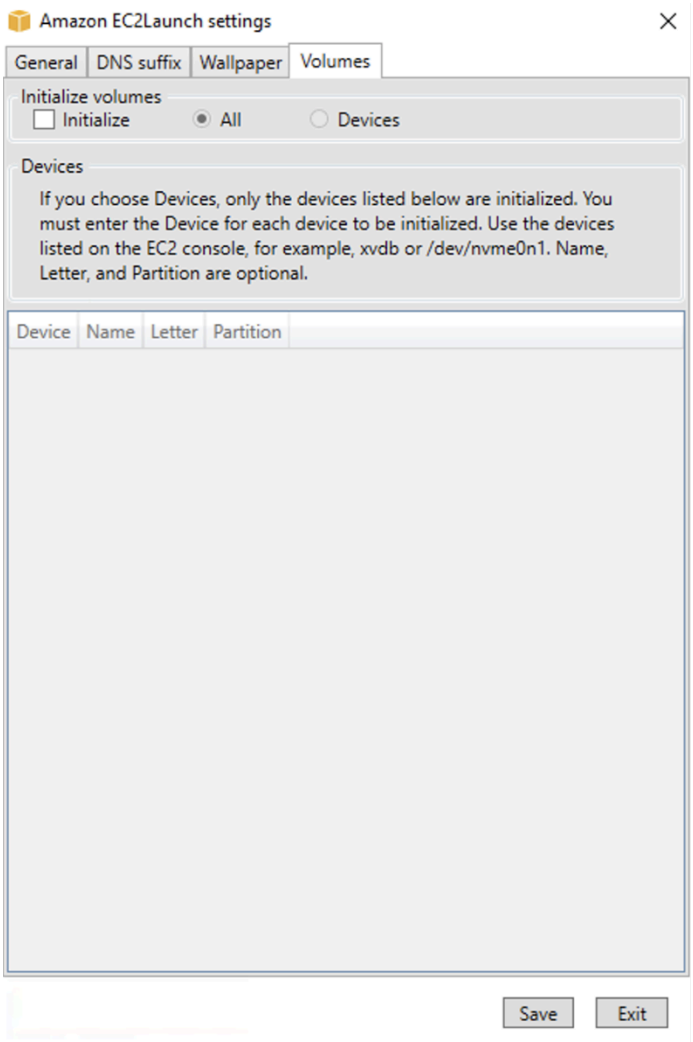
> **Note**
>
> Anda harus mengaktifkan tanda dalam metadata untuk menampilkan tanda pada wallpaper. Untuk informasi selengkapnya tentang tanda instans dan metadata, lihat Melihat tag untuk instans EC2 menggunakan metadata instans.




6. Pada tab Volume, pilih apakah Anda ingin menginisialisasi volume yang dilampirkan ke instans. Mengaktifkan set huruf drive untuk volume tambahan dan memperluasnya untuk menggunakan ruang yang tersedia. Jika Anda memilih Semua, semua volume penyimpanan diinisialisasi. Jika Anda memilih Perangkat, hanya perangkat yang ditentukan dalam daftar yang diinisialisasi. Anda harus memasukkan perangkat untuk setiap perangkat yang akan diinisialisasi. Gunakan peranti yang terdaftar pada konsol EC2, sebagai contoh, `xvdb` atau `/dev/nvme0n1`. Daftar dropdown

menampilkan volume penyimpanan yang dilampirkan pada instans. Untuk memasukkan perangkat yang tidak terpasang ke instans, masukkan perangkat itu di bidang teks.

Nama, Huruf, dan Partisi adalah bidang opsional. Jika tidak ada nilai yang ditentukan untuk Partisi, volume penyimpanan yang lebih besar dari 2 TB diinisialisasi dengan tipe `gpt` partisi, dan volume yang lebih kecil dari 2 TB diinisialisasi dengan tipe `mbr` partisi. Jika perangkat dikonfigurasi, dan perangkat non-NTFS berisi tabel partisi, atau 4 KB pertama dari disk berisi data, maka disk akan dilewati dan tindakan dicatat.




## Konfigurasikan EC2Launch v2 menggunakan CLI

Anda dapat menggunakan Command Line Interface (CLI) untuk mengonfigurasi pengaturan EC2Launch Anda dan mengelola layanan. Bagian berikut ini berisi deskripsi dan informasi penggunaan untuk perintah CLI yang dapat Anda gunakan untuk mengelola EC2Launch v2.

> **Note**
>
> EC2launch v2 CLI yang dapat dieksekusi,`EC2Launch.exe`, berada di. `C:\Program Files\Amazon\EC2Launch\` Secara default, direktori ini tidak ada di Windows PATH. `ec2launch` *`command`*Bentuk yang ditunjukkan dalam contoh di bagian ini adalah singkatan yang tidak dikenali oleh shell secara langsung. Untuk menjalankan perintah, gunakan jalur lengkap ke yang dapat dieksekusi, atau pertama-tama ubah ke direktori instalasi. PowerShell Contoh berikut menggunakan jalur penuh dengan operator panggilan:
>
> ```
> & "C:\Program Files\Amazon\EC2Launch\EC2Launch.exe" status
> ```

Commands

- collect-logs
- get-agent-config
- list-volumes
- atur ulang
- run
- status
- sysprep
- validasi
- versi
- wallpaper

collect-logs

Mengumpulkan file log untuk EC2Launch, men-zip file, dan menempatkannya di direktori yang ditentukan.

Contoh

```
ec2launch collect-logs -o C:\Mylogs.zip
```

Penggunaan

`ec2launch collect-logs [flags]`

Bendera

`-h, --help`

bantuan untuk `collect-logs`

`-o, --output string`

jalur ke file log output zip

get-agent-config

Mencetak `agent-config.yml` dalam format yang ditentukan (JSON atau YAML). Jika tidak ada format yang ditentukan, `agent-config.yml` dicetak dalam format yang ditentukan sebelumnya.

Contoh

```
ec2launch get-agent-config -f json
```

Penggunaan

`ec2launch get-agent-config [flags]`

Bendera

`-h, --help`

bantuan untuk `get-agent-config`

`-f, --format string`

format output file `agent-config`: `json`, `yaml`

list-volumes

Mencantumkan semua volume penyimpanan yang dilampirkan ke instans, termasuk volume singkat dan EBS.

Contoh

```
ec2launch list-volumes
```

Penggunaan

`ec2launch list-volumes`

Bendera

`-h, --help`

bantuan untuk `list-volumes`

atur ulang

Tujuan utama dari tugas ini adalah untuk mengatur ulang agen untuk waktu berikutnya yang dijalankan. Untuk melakukan itu, perintah reset menghapus semua data status agen untuk EC2launch v2 dari direktori `EC2Launch` lokal (lihat Struktur direktori EC2Launch v2). Reset opsional menghapus layanan dan log Sysprep.

Perilaku skrip tergantung pada mode apa agen menjalankan skrip — inline, atau terpisah.

Inline (default)

Agen EC2Launch v2 menjalankan skrip satu per satu (`detach: false`) Ini adalah pengaturan default.

> **Note**
>
> Ketika skrip inline Anda mengeluarkan perintah reset atau sysprep, skrip tersebut segera berjalan dan mengatur ulang agen. Tugas saat ini selesai, kemudian agen dimatikan tanpa menjalankan tugas lebih lanjut.
> Misalnya, jika tugas yang mengeluarkan perintah akan diikuti oleh `startSsm` tugas (disertakan secara default setelah data pengguna berjalan), tugas tidak berjalan dan layanan Systems Manager tidak pernah dimulai.

Terlepas

Agen EC2launch v2 menjalankan skrip bersamaan dengan tugas lain (`detach: true`).

> **Note**
>
> Saat skrip terpisah Anda mengeluarkan reset atau sysprep, perintah tersebut menunggu agen selesai sebelum dijalankan. Tugas setelah executeScript masih akan berjalan.

Contoh

```
ec2launch reset -c
```

Penggunaan

`ec2launch reset [flags]`

Bendera

`-c`, `--clean`

membersihkan log instans sebelum `reset`

`-h`, `--help`

bantuan untuk `reset`

run

Jalankan EC2Launch v2.

Contoh

```
ec2launch run
```

Penggunaan

`ec2launch run [flags]`

Bendera

`-h`, `--help`

bantuan untuk `run`

status

Dapatkan status agen EC2Launch v2. Memblokir proses secara opsional sampai agen selesai. Kode keluar proses menentukan status agen:

- `0` – agen berjalan dan berhasil.
- `1` – agen berjalan dan gagal.
- `2` – agen masih berjalan.
- `3` – agen dalam status yang tidak diketahui. Status agen tidak berjalan atau berhenti.
- `4` – kesalahan terjadi ketika mencoba untuk mengambil status agen.
- `5` – agen tidak berjalan dan status berjalan terakhir yang diketahui tidak diketahui. Ini bisa berarti salah satu dari berikut ini:
  - kedua `state.json` dan `previous-state.json` dihapus.
  - `previous-state.json` rusak.

  Ini adalah status agen setelah menjalankan perintah `reset`.

Contoh:

```
ec2launch status -b
```

> **Note**
>
> Per `status` intah menulis pesan singkat ke konsol (misalnya,`agent ran successfully`). Namun, nilai kode keluar proses yang tercantum sebelumnya menentukan status agen, bukan pesan itu. Untuk melihat kode keluar PowerShell, baca variabel `$LASTEXITCODE` otomatis setelah Anda menjalankan perintah, misalnya:
>
> ```
> & "C:\Program Files\Amazon\EC2Launch\EC2Launch.exe" status; "ExitCode=
> $LASTEXITCODE"
> ```
>
> Output `ExitCode=0` menunjukkan bahwa agen berjalan dengan sukses.

Penggunaan

`ec2launch status [flags]`

Bendera

`-b,--block`

memblokir proses sampai agen selesai berjalan

`-h,--help`

bantuan untuk `status`

sysprep

Tujuan utama dari tugas ini adalah untuk mengatur ulang agen untuk waktu berikutnya yang dijalankan. Untuk melakukan itu, perintah sysprep mengatur ulang status agen, memperbarui file `unattend.xml`, menonaktifkan RDP, dan menjalankan Sysprep.

Perilaku skrip tergantung pada mode apa agen menjalankan skrip — inline, atau terpisah.

Inline (default)

Agen EC2Launch v2 menjalankan skrip satu per satu (`detach: false`) Ini adalah pengaturan default.

> **Note**
>
> Ketika skrip inline Anda mengeluarkan perintah reset atau sysprep, skrip tersebut segera berjalan dan mengatur ulang agen. Tugas saat ini selesai, kemudian agen dimatikan tanpa menjalankan tugas lebih lanjut.
> Misalnya, jika tugas yang mengeluarkan perintah akan diikuti oleh `startSsm` tugas (disertakan secara default setelah data pengguna berjalan), tugas tidak berjalan dan layanan Systems Manager tidak pernah dimulai.

Terlepas

Agen EC2launch v2 menjalankan skrip bersamaan dengan tugas lain (`detach: true`).

> **Note**
>
> Saat skrip terpisah Anda mengeluarkan reset atau sysprep, perintah tersebut menunggu agen selesai sebelum dijalankan. Tugas setelah executeScript masih akan berjalan.

Contoh:

```
ec2launch sysprep
```

Penggunaan

`ec2launch sysprep [flags]`

Bendera

`-c,--clean`

membersihkan log instans sebelum `sysprep`

`-h,--help`

bantuan untuk Sysprep

`-s,--shutdown`

mematikan instans setelah `sysprep`

validasi

Memvalidasi `agent-config` file `C:\ProgramData\Amazon\EC2Launch\config\agent-config.yml`.

Contoh

```
ec2launch validate
```

Penggunaan

`ec2launch validate [flags]`

Bendera

`-h , --help`

bantuan untuk `validate`

versi

Mendapatkan versi yang dapat dieksekusi.

Contoh

```
ec2launch version
```

Penggunaan

`ec2launch version [flags]`

Bendera

`-h`, `--help`

bantuan untuk `version`

wallpaper

Menyetel wallpaper baru ke jalur wallpaper yang disediakan (file.jpg), dan menampilkan detail instans yang dipilih.

Sintaksis

```
ec2launch wallpaper ^
--path="C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg" ^
--all-tags ^
--
attributes=hostName,instanceId,privateIpAddress,publicIpAddress,ipv6Address,instanceSize,availa
```

Masukan

Parameter

--tag yang diizinkan [,] ***`tag-name-1 tag-name-n`***

(Opsional) Base64 mengenkode array JSON dari nama tanda instans untuk ditampilkan di wallpaper. Anda dapat menggunakan tanda ini atau `--all-tags`, tetapi tidak keduanya.

--atribut***`attribute-string-1`***, ***`attribute-string-n`***

(Opsional) Daftar string atribut `wallpaper` yang dipisahkan dengan koma untuk menerapkan pengaturan ke wallpaper.

[--jalur | -p] ***path-string***

(Wajib) Menentukan jalur file gambar latar belakang `wallpaper`.

Bendera

--all-tags

(Opsional) Menampilkan semua tanda instans pada wallpaper. Anda dapat menggunakan tanda ini atau `--allowed-tags`, tetapi tidak keduanya.

[--help | -h]

Menampilkan bantuan untuk perintah wallpaper.

Konfigurasi EC2Launch v2

Bagian ini mencakup skema, tugas, detail, dan contoh konfigurasi untuk `agent-config.yml` dan data pengguna.

Tugas dan contoh

- Skema: agent-config.yml
- Konfigurasikan skrip data pengguna EC2launch v2 yang berjalan selama peluncuran atau reboot

Skema: **`agent-config.yml`**

Struktur `agent-config.yml` file ditunjukkan di bawah ini. Perhatikan bahwa tugas tidak dapat diulang dalam tahap yang sama. Untuk properti tugas, lihat deskripsi tugas yang mengikuti.

Struktur dokumen: agent-config.yml

JSON

```
{
 "version": "1.1",
 "config": [
  {
   "stage": "string",
   "tasks": [
    {
```

```
     "task": "string",
     "inputs": {
      ...
     }
    },
    ...
   ]
  },
  ...
 ]
}
```

YAML

```
version: 1.1
config:
- stage: string
  tasks:
  - task: string
 inputs:
   ...
  ...
...
```

Contoh: **`agent-config.yml`**

Contoh berikut menunjukkan pengaturan untuk file konfigurasi `agent-config.yml`.

```
version: 1.1
config:
- stage: boot
  tasks:
  - task: extendRootPartition
- stage: preReady
  tasks:
  - task: activateWindows
    inputs:
      activation:
        type: amazon
  - task: setDnsSuffix
    inputs:
      suffixes:
      - $REGION.ec2-utilities.amazonaws.com
```

```
  - task: setAdminAccount
    inputs:
      password:
        type: random
  - task: setWallpaper
    inputs:
      path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
      attributes:
      - hostName
      - instanceId
      - privateIpAddress
      - publicIpAddress
      - instanceSize
      - availabilityZone
      - architecture
- stage: postReady
  tasks:
  - task: startSsm
```

Konfigurasikan skrip data pengguna EC2launch v2 yang berjalan selama peluncuran atau reboot

Contoh JSON dan YAMG berikut menunjukkan struktur dokumen untuk data pengguna. Amazon EC2 mengurai setiap tugas yang dinamai dalam array `tasks` yang Anda tentukan dalam dokumen. Setiap tugas memiliki set properti dan persyaratan sendiri. Untuk detailnya, lihat Definisi tugas untuk tugas startup EC2launch v2.

> **Note**
>
> Tugas hanya boleh muncul sekali dalam array tugas data pengguna.

Struktur dokumen: data pengguna

JSON

```
{
 "version": "1.1",
 "tasks": [
  {
   "task": "string",
   "inputs": {
    ...
```

```
   },
  },
  ...
 ]
}
```

YAML

```
version: 1.1
tasks:
- task: string
  inputs:
    ...
...
```

Contoh: data pengguna

Untuk informasi selengkapnya tentang data pengguna, lihat Cara Amazon EC2 menangani data pengguna untuk instans Windows.

Contoh dokumen YAML berikut menunjukkan PowerShell skrip yang EC2launch v2 jalankan sebagai data pengguna untuk membuat file.

```
version: 1.1
tasks:
- task: executeScript
  inputs:
  - frequency: always
    type: powershell
    runAs: localSystem
    content: |-
      New-Item -Path 'C:\PowerShellTest.txt' -ItemType File
```

Anda dapat menggunakan format XML untuk data pengguna yang kompatibel dengan versi agen peluncuran sebelumnya. EC2Launch v2 menjalankan skrip sebagai tugas `executeScript` di tahap `UserData`. Untuk menyesuaikan dengan perilaku EC2launch v1 dan EC2config, skrip data pengguna berjalan sebagai attached/inline proses secara default.

Anda dapat menambahkan tanda opsional untuk menyesuaikan cara skrip Anda berjalan. Misalnya, untuk menjalankan skrip data pengguna saat instans di-boot ulang selain satu kali saat instans diluncurkan, Anda dapat menggunakan tanda berikut:

`<persist>true</persist>`

Contoh:

```
<powershell>
  $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
  New-Item $file -ItemType file
</powershell>
<persist>true</persist>
```

Anda dapat menentukan satu atau lebih PowerShell argumen dengan `<powershellArguments>` tag. Jika tidak ada argumen yang diteruskan, EC2launch v2 menambahkan argumen berikut secara default:. `-ExecutionPolicy Unrestricted`

Contoh:

```
<powershell>
  $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
  New-Item $file -ItemType file
</powershell>
<powershellArguments>-ExecutionPolicy Unrestricted -NoProfile -NonInteractive</
powershellArguments>
```

Untuk menjalankan skrip data pengguna XML sebagai proses yang terpisah, tambahkan tanda berikut ke data pengguna Anda.

`<detach>true</detach>`

Contoh:

```
<powershell>
  $file = $env:SystemRoot + "\Temp" + (Get-Date).ToString("MM-dd-yy-hh-mm")
  New-Item $file -ItemType file
</powershell>
<detach>true</detach>
```

> **Note**
>
> Tanda lepas tidak didukung pada agen peluncuran sebelumnya.

Log perubahan: data pengguna

Tabel berikut mencantumkan perubahan untuk data pengguna, dan referensi silang ke versi agen EC2launch v2 yang berlaku.

| Versi data pengguna | Detail | Diperkenalkan di |
|---|---|---|
| 1.1 | • Tugas data pengguna berjalan sebelum tahap `PostReady` dalam file konfigurasi agen.<br>• Menjalankan data pengguna sebelum memulai Systems Manager Agent (perilaku yang sama seperti EC2Launch v1 dan EC2Config).* | EC2Launch v2 versi 2.0.1245 |
| 1.0 | • Akan usang.<br>• Tugas data pengguna berjalan sebelum tahap `PostReady` dalam file konfigurasi agen. Ini tidak kompatibel dengan EC2Launch v1.<br>• Dipengaruhi oleh kondisi balapan antara start Systems Manager Agent dan tugas data pengguna. | EC2Launch v2 versi 2.0.0 |

* Bila digunakan dengan file `agent-config.yml` default.

Kode keluar dan boot ulang EC2Launch v2

Anda dapat menggunakan EC2Launch v2 untuk menentukan bagaimana kode keluar ditangani oleh skrip Anda. Secara default, kode keluar dari perintah terakhir yang dijalankan dalam skrip dilaporkan sebagai kode keluar untuk seluruh skrip. Sebagai contoh, jika skrip mencakup tiga perintah dan perintah pertama gagal tetapi perintah yang berikutnya berhasil, maka status berjalan dilaporkan sebagai `success` karena perintah akhir berhasil.

Jika Anda ingin skrip me-reboot sebuah instans, maka Anda harus menentukan `exit 3010` dalam skrip Anda, bahkan saat reboot menjadi langkah terakhir dalam skrip Anda. `exit 3010` menginstruksikan EC2Launch v2 untuk me-reboot instans dan memanggil skrip lagi sampai

mengembalikan kode keluar yang tidak, `3010`atau sehingga jumlah reboot maksimum telah dicapai. EC2Launch v2 memungkinkan maksimal 5 boot ulang per tugas. Jika Anda mencoba untuk me-reboot instans dari skrip dengan menggunakan mekanisme yang berbeda, seperti, `Restart-Computer`maka status berjalan skrip akan menjadi tidak konsisten. Sebagai contoh, skrip mungkin terjebak dalam loop mulai ulang atau tidak melakukan restart.

Jika Anda menggunakan format data pengguna XMLyang kompatibel dengan agen sebelumnya, maka data pengguna dapat berjalan lebih banyak daripada yang Anda inginkan. Untuk informasi selengkapnya, lihat Layanan menjalankan data pengguna lebih dari satu kali di bagian Pemecahan Masalah.

EC2Launch v2 dan Sysprep

Layanan EC2Launch v2 menjalankan Sysprep, alat Microsoft yang memungkinkan Anda untuk membuat AMI Windows kustom yang dapat digunakan kembali. Saat EC2Launch v2 memanggil Sysprep, file digunakan di `%ProgramData%\Amazon\EC2Launch` untuk menentukan operasi mana yang akan dilakukan. Anda dapat mengedit file ini secara tidak langsung menggunakan kotak dialog pengaturan EC2Launch, atau langsung menggunakan editor YAML atau editor teks. Namun, ada beberapa pengaturan lanjutan yang tidak tersedia di kotak dialog pengaturan EC2Launch, jadi Anda harus mengedit entri tersebut secara langsung.

Jika Anda membuat AMI dari sebuah instans setelah memperbarui pengaturannya, pengaturan baru tersebut diterapkan ke setiap instans yang diluncurkan dari AMI baru. Untuk informasi tentang membuat grafik, lihat Buat EBS-backed AMI Amazon.

Definisi tugas untuk tugas startup EC2launch v2

Setiap tugas yang dijalankan EC2launch v2 selama peluncuran atau startup memiliki serangkaian properti dan persyaratannya sendiri. Detail tugas mencakup pengaturan untuk seberapa sering tugas berjalan — sekali, atau selalu, tahap apa dari proses boot agen yang dijalankannya, sintaks, dan contoh dokumen YAML. Untuk informasi selengkapnya, tinjau detail tugas yang ditunjukkan dalam referensi ini.

Tugas EC2launch v2

- activateWindows
- mengaktifkan JumboFrames
- mengaktifkan OpenSsh
- executeProgram

- executeScript
- memperpanjang RootPartition
- initializeVolume
- optimizeEna
- set AdminAccount
- set DnsSuffix
- set HostName
- setWallpaper
- startSsm
- sysprep
- writeFile

activateWindows

Mengaktifkan Windows terhadap satu set AWS KMS server. Aktivasi dilewati jika instance terdeteksi sebagai Bring-Your-Own-License (BYOL).

Frekuensi - sekali

AllowedStages — `[PreReady]`

Masukan —

`activation`: (peta)

`type`: (string) tipe aktivasi yang akan digunakan, diatur ke `amazon`

Contoh

```
task: activateWindows
  inputs:
    activation:
    type: amazon
```

mengaktifkan JumboFrames

Mengaktifkan Jumbo Frames, yang meningkatkan unit transmisi maksimum (MTU) dari adaptor jaringan. Untuk informasi selengkapnya, lihat Frame jumbo (9001 MTU).

Frekuensi - selalu

AllowedStages — `[PostReady, UserData]`

Masukan - tidak ada

Contoh

```
task: enableJumboFrames
```

mengaktifkan OpenSsh

Mengaktifkan Windows OpenSSH dan menambahkan kunci publik untuk instans tersebut ke folder kunci resmi.

Frekuensi — sekali

AllowedStages — `[PreReady, UserData]`

Masukan - tidak ada

Contoh

Contoh berikut menunjukkan cara mengaktifkan OpenSSH pada sebuah instans, dan untuk menambahkan kunci publik untuk instans tersebut ke folder kunci resmi. Konfigurasi ini hanya berfungsi pada instans yang menjalankan Windows Server 2019 dan versi setelahnya.

```
task: enableOpenSsh
```

executeProgram

Menjalankan program dengan argumen opsional dan frekuensi tertentu.

Tahapan: Anda dapat menjalankan tugas `executeProgram` selama tahapan `PreReady`, `PostReady`, dan `UserData`

Frekuensi: dapat dikonfigurasi, lihat Input.

Masukan

Bagian ini berisi satu atau lebih program untuk menjalankan executeProgram tugas (input). Setiap input dapat mencakup pengaturan yang dapat dikonfigurasi berikut:

frekuensi (string)

(Wajib) Tentukan dengan tepat satu dari nilai-nilai berikut:

- `once`
- `always`

jalur (string)

(Wajib) Jalur file untuk menjalankan executable.

argumen (daftar string)

(Opsional) Daftar argumen yang dipisahkan koma untuk diberikan kepada program sebagai input.

runAs (string)

(Wajib) Harus diatur ke `localSystem`

Output

Semua tugas menulis entri logfile ke file `agent.log`. Output tambahan dari tugas `executeProgram` disimpan secara terpisah dalam folder bernama dinamis, sebagai berikut:

`%LocalAppData%\Temp\`*`EC2Launch#########\outputfilename.tmp`*

Jalur yang tepat ke file output disertakan dalam `agent.log` file, misalnya:

```
Program file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\ExecuteProgramInputs.tmp
Output file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Output.tmp
Error file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Err.tmp
```

File keluaran untuk tugas **`ExecuteProgram`**

**`ExecuteProgramInputs.tmp`**

Berisi jalur untuk executable, dan semua parameter input yang diteruskan tugas `executeProgram` padanya saat dijalankan.

**`Output.tmp`**

Berisi output runtime dari program yang dijalankan tugas `executeProgram`.

**`Err.tmp`**

Berisi pesan kesalahan runtime dari program yang dijalankan tugas `executeProgram`.

Contoh

Contoh berikut menunjukkan cara menjalankan file yang dapat dieksekusi dari direktori lokal pada instans dengan tugas `executeProgram`.

Contoh 1: File setup yang dapat dieksekusi dengan satu argumen

Contoh ini menunjukkan tugas `executeProgram` yang menjalankan setup yang dapat dieksekusi dalam mode senyap.

```
task: executeProgram
  inputs:
    - frequency: always
      path: C:\Users\Administrator\Desktop\setup.exe
      arguments: ['-quiet']
```

Contoh 2: VLC dapat dieksekusi dengan dua argumen

Contoh ini menunjukkan tugas `executeProgram` yang menjalankan file VLC yang dapat dieksekusi dengan dua argumen yang diteruskan sebagai parameter input.

```
task: executeProgram
  inputs:
    - frequency: always
      path: C:\vlc-3.0.11-win64.exe
      arguments: ['/L=1033','/S']
      runAs: localSystem
```

executeScript

Menjalankan skrip dengan argumen opsional dan frekuensi tertentu. Perilaku skrip tergantung pada mode apa agen menjalankan skrip — inline, atau terpisah.

Inline (default)

Agen EC2Launch v2 menjalankan skrip satu per satu (`detach: false`) Ini adalah pengaturan default.

> **Note**
>
> Ketika skrip inline Anda mengeluarkan perintah reset atau sysprep, skrip tersebut segera berjalan dan mengatur ulang agen. Tugas saat ini selesai, kemudian agen dimatikan tanpa menjalankan tugas lebih lanjut.
> Misalnya, jika tugas yang mengeluarkan perintah akan diikuti oleh `startSsm` tugas (disertakan secara default setelah data pengguna berjalan), tugas tidak berjalan dan layanan Systems Manager tidak pernah dimulai.

Terlepas

Agen EC2launch v2 menjalankan skrip bersamaan dengan tugas lain (`detach: true`).

> **Note**
>
> Saat skrip terpisah Anda mengeluarkan reset atau sysprep, perintah tersebut menunggu agen selesai sebelum dijalankan. Tugas setelah executeScript masih akan berjalan.

Tahapan: Anda dapat menjalankan tugas `executeScript` selama tahapan `PreReady`, `PostReady`, dan `UserData`

Frekuensi: dapat dikonfigurasi, lihat Input.

Masukan

Bagian ini berisi satu atau lebih skrip untuk executeScript tugas yang akan dijalankan (input). Setiap input dapat mencakup pengaturan yang dapat dikonfigurasi berikut:

frekuensi (string)

(Wajib) Tentukan dengan tepat satu dari nilai-nilai berikut:

- `once`
- `always`

tipe (string)

(Wajib) Tentukan dengan tepat satu dari nilai-nilai berikut:

- `batch`

- `powershell`

argumen (daftar string)

(Opsional) Daftar argumen string untuk diteruskan ke shell (bukan ke PowerShell skrip). Parameter ini tidak didukung untuk `type: batch`. Jika tidak ada argumen yang diteruskan, EC2launch v2 menambahkan argumen berikut secara default:. `-ExecutionPolicy Unrestricted`

konten (string)

(Wajib) Konten skrip.

runAs (string)

(Wajib) Tentukan dengan tepat satu dari nilai-nilai berikut:

- `admin`
- `localSystem`

lepas (Boolean)

(Opsional) Agen EC2launch v2 diatur default untuk menjalankan skrip satu per satu (`detach: false`). Untuk menjalankan skrip secara bersamaan dengan tugas lain, atur nilainya ke `true` (`detach: true`).

> **Note**
>
> Kode keluar skrip (termasuk `3010`) tidak berpengaruh jika `detach` diatur ke `true`.

Output

Semua tugas menulis entri logfile ke file `agent.log`. Output tambahan dari skrip yang dijalankan tugas `executeScript` disimpan secara terpisah dalam folder bernama dinamis, sebagai berikut:

`%LocalAppData%\Temp\`*`EC2Launch#########\outputfilename.ext`*

Jalur yang tepat ke file output disertakan dalam `agent.log` file, misalnya:

```
Program file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\UserScript.ps1
Output file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Output.tmp
```

```
Error file is created at: C:\Windows\system32\config\systemprofile\AppData\Local
\Temp\EC2Launch123456789\Err.tmp
```

File keluaran untuk tugas **`ExecuteScript`**

**`UserScript.`*`ext`***

Berisi skrip yang dijalankan tugas `executeScript`. Ekstensi file tergantung pada jenis skrip yang Anda tentukan dalam `type` parameter untuk `executeScript` tugas, sebagai berikut:

- Jika tipenya adalah `batch`, maka ekstensi file adalah `.bat`.
- Jika tipenya adalah `powershell`, maka ekstensi file adalah `.ps1`.

**`Output.tmp`**

Berisi output runtime dari skrip yang dijalankan tugas `executeScript`.

**`Err.tmp`**

Berisi pesan kesalahan runtime dari skrip yang dijalankan tugas `executeScript`.

Contoh

Contoh berikut menunjukkan cara menjalankan skrip inline dengan tugas `executeScript`.

Contoh 1: File teks output Hello world

Contoh ini menunjukkan `executeScript` tugas yang menjalankan PowerShell skrip untuk membuat file teks “Hello world” pada `C:` drive.

```
task: executeScript
  inputs:
    - frequency: always
      type: powershell
      runAs: admin
      content: |-
        New-Item -Path 'C:\PowerShellTest.txt' -ItemType File
        Set-Content 'C:\PowerShellTest.txt' "Hello world"
```

Contoh 2: Jalankan dua skrip

Contoh ini menunjukkan bahwa tugas `executeScript` dapat menjalankan lebih dari satu skrip, dan tipe skrip tidak harus cocok.

Script pertama (`type: powershell`) menulis ringkasan proses yang saat ini berjalan pada instans ke file teks yang terletak di `C:` drive.

Script kedua (`batch`) menulis informasi sistem ke `Output.tmp` file.

```
task: executeScript
  inputs:
    - frequency: always
      type: powershell
      runAs: localSystem
      content: |
        Get-Process | Out-File -FilePath C:\Process.txt
    - frequency: always
      type: batch
      runAs: localSystem
      content: |
        systeminfo
```

Contoh 3: Konfigurasi sistem idempotensi dengan boot ulang

Contoh ini menunjukkan tugas `executeScript` yang menjalankan skrip idempotensi untuk melakukan konfigurasi sistem berikut dengan boot ulang di antara setiap langkah:

- Ganti nama komputer.
- Bergabunglah dengan komputer ke domain.
- Aktifkan Telnet.

Skrip memastikan bahwa setiap operasi berjalan satu kali saja. Ini mencegah loop reboot dan membuat skrip idempoten.

```
task: executeScript
  inputs:
    - frequency: always
      type: powershell
      runAs: localSystem
      content: |-
        $name = $env:ComputerName
        if ($name -ne $desiredName) {
          Rename-Computer -NewName $desiredName
          exit 3010
        }
        $domain = Get-ADDomain
```

```
        if ($domain -ne $desiredDomain)
        {
          Add-Computer -DomainName $desiredDomain
          exit 3010
        }
        $telnet = Get-WindowsFeature -Name Telnet-Client
        if (-not $telnet.Installed)
        {
          Install-WindowsFeature -Name "Telnet-Client"
          exit 3010
        }
```

memperpanjang RootPartition

Memperluas volume root untuk menggunakan semua ruang yang tersedia di disk.

Frekuensi - sekali

AllowedStages — `[Boot]`

Masukan - tidak ada

Contoh

```
task: extendRootPartition
```

initializeVolume

Menginisialisasi volume kosong yang dilampirkan ke instans sehingga mereka diaktifkan dan dipartisi. Agen peluncuran melewatkan inisialisasi jika mendeteksi bahwa volume tidak kosong. Volume dianggap kosong jika 4 KiB pertama volume kosong, atau jika volume tidak memiliki tata letak Windows-recognizable drive.

Parameter `letter` input selalu diterapkan saat tugas ini berjalan, terlepas dari apakah drive sudah diinisialisasi.

Tugas `initializeVolume` melakukan tindakan berikut.

- Atur atribut disk `offline` dan `readonly` ke false.
- Buat sebuah partisi. Jika tidak ada jenis partisi yang ditentukan dalam parameter `partition` input, default berikut berlaku:

  - Jika ukuran disk lebih kecil dari 2 TB, atur tipe partisi ke `mbr`.
  - Jika ukuran disk 2 TB atau lebih besar, atur tipe partisi ke `gpt`.
- Format volume sebagai NTFS.
- Atur label volume sebagai berikut:
  - Gunakan nilai parameter input `name`, jika ditentukan.
  - Jika volumenya fana, dan tidak ada nama yang ditentukan, atur label volume ke. `Temporary Storage Z`
- Jika volumenya singkat (SSD atau HDD - bukan Amazon EBS), buat `Important.txt` file di root volume dengan konten berikut:

```
This is an 'Instance Store' disk and is provided at no additional charge.

*This disk offers increased performance since it is local to the host
*The number of Instance Store disks available to an instance vary by instance type
*DATA ON THIS DRIVE WILL BE LOST IN CASES OF IMPAIRMENT OR STOPPING THE INSTANCE.
 PLEASE ENSURE THAT ANY IMPORTANT DATA IS BACKED UP FREQUENTLY

For more information, please refer to: Penyimpanan blok sementara instans untuk
 instans EC2.
```

- Atur huruf drive ke nilai yang ditentukan dalam parameter `letter` input.

Tahapan: Anda dapat menjalankan `initializeVolume` tugas selama `PostReady` dan `UserData` tahapan.

Frekuensi: selalu.

Masukan

Anda dapat mengonfigurasi parameter runtime sebagai berikut:

perangkat (daftar peta)

(Bersyarat) Konfigurasi untuk setiap perangkat yang dimulai agen peluncuran. Ini diperlukan jika parameter input `initialize` diatur ke `devices`.

- perangkat (string, wajib) - Mengidentifikasi perangkat selama pembuatan instans. Sebagai contoh, `xvdb`, `xvdf`, atau `\dev\nvme0n1`.
- huruf (string, opsional) - Satu karakter. Surat drive untuk ditetapkan.

- nama (string, opsional) - Nama volume yang akan ditetapkan.
- partisi (string, opsional) – Tentukan salah satu nilai berikut untuk tipe partisi yang akan dibuat, atau biarkan agen peluncuran menentukan default berdasarkan ukuran volume:
  - Mbr
  - gpt

inisialisasi (string)

(Wajib) Tentukan dengan tepat satu dari nilai-nilai berikut:

- `all`
- `devices`

Contoh

Contoh berikut menampilkan konfigurasi input sampel untuk tugas `initializeVolume` tersebut.

Contoh 1: Inisialisasi dua volume pada sebuah instans

Contoh ini menunjukkan tugas `initializeVolume` yang menginisialisasi dua volume sekunder pada sebuah instans. Perangkat yang bernama `DataVolume2` dalam contoh tersebut bersifat sementara.

```
task: initializeVolume
inputs:
  initialize: devices
  devices:
  - device: xvdb
    name: DataVolume1
    letter: D
    partition: mbr
  - device: /dev/nvme0n1
    name: DataVolume2
    letter: E
    partition: gpt
```

Contoh 2: Inisialisasi volume EBS yang dilampirkan ke sebuah instans

Contoh ini menunjukkan tugas `initializeVolume` yang menginisialisasi semua volume EBS kosong yang dilampirkan ke instans.

```
task: initializeVolume
inputs:
  initialize: all
```

optimizeEna

Mengoptimalkan pengaturan ENA berdasarkan tipe instans saat ini; mungkin mem-boot ulang instans.

Frekuensi - selalu

AllowedStages — `[PostReady, UserData]`

Masukan - tidak ada

Contoh

```
task: optimizeEna
```

set AdminAccount

Set atribut untuk akun administrator default yang dibuat di mesin lokal.

Frekuensi - sekali

AllowedStages — `[PreReady]`

Masukan —

`name`: (string) nama akun administrator

`password`: (peta)

`type`: (string) strategi untuk mengatur kata sandi, baik sebagai `static`, `random`, atau `doNothing`

`data`: (string) menyimpan data jika bidang `type` statis

Contoh

```
task: setAdminAccount
inputs:
  name: Administrator
  password:
```

```
  type: random
```

set DnsSuffix

Menambahkan sufiks DNS ke daftar sufiks pencarian. Hanya sufiks yang belum ada yang ditambahkan ke daftar. Untuk informasi selengkapnya tentang cara agen peluncuran mengatur sufiks DNS, lihat. Konfigurasikan Sufiks DNS untuk agen peluncuran EC2 Windows

Frekuensi - selalu

AllowedStages — `[PreReady]`

Masukan —

`suffixes`: (daftar string) daftar satu atau lebih sufiks DNS yang valid; variabel substitusi yang valid adalah `$REGION` dan `$AZ`

Contoh

```
task: setDnsSuffix
inputs:
  suffixes:
  - $REGION.ec2-utilities.amazonaws.com
```

set HostName

Menetapkan nama host komputer menjadi string kustom atau, jika `hostName` tidak ditentukan, alamat IPv4 privat-nya.

Frekuensi — selalu

AllowedStages — `[PostReady, UserData]`

Masukan —

`hostName`: (string) nama host opsional, yang harus diformat sebagai berikut.

- Harus 15 karakter atau kurang
- Harus berisi hanya karakter alfanumerik (a-z, A-Z, 0-9) dan tanda hubung (-).
- Tidak boleh seluruhnya terdiri dari karakter numerik.

`reboot`: (boolean) menunjukkan apakah booti ulang diizinkan saat nama host diubah

Contoh

```
task: setHostName
inputs:
  reboot: true
```

setWallpaper

Membuat file pintasan `setwallpaper.lnk` di folder startup setiap pengguna yang ada kecuali untuk `Default User`. File pintasan ini berjalan saat pengguna masuk untuk pertama kalinya setelah boot instans. File ini menyiapkan instans dengan wallpaper kustom yang menampilkan atribut instans.

Jalur file pintasan adalah:

```
$env:SystemDrive/Users/<user>/AppData/Roaming/Microsoft/Windows/Start Menu/Programs/
Startup/setwallpaper.lnk
```

> **Note**
>
> Saat Anda menghapus tugas `setWallpaper`, file pintasan ini tidak akan terhapus. Untuk informasi selengkapnya, lihat Tugas setWallpaper tidak diaktifkan tetapi wallpaper diatur ulang saat reboot.

Tahapan: Anda dapat mengonfigurasi wallpaper selama tahapan `PreReady` dan `UserData`.

Frekuensi: `always`

Konfigurasi wallpaper

Anda dapat menggunakan pengaturan berikut untuk mengonfigurasi wallpaper Anda.

Masukan

Parameter masukan yang Anda berikan, dan atribut yang dapat Anda atur untuk mengonfigurasi wallpaper Anda:

jalur (string)

(Wajib) Jalur nama file dari file gambar format .jpg lokal yang akan digunakan untuk gambar wallpaper Anda.

atribut (daftar string)

(Opsional) Anda dapat menambahkan satu atau lebih atribut berikut ke wallpaper Anda:

- `architecture`
- `availabilityZone`
- `hostName`
- `instanceId`
- `instanceSize`
- `privateIpAddress`
- `publicIpAddress`
- `ipv6Address`

instanceTags

(Opsional) Anda dapat menggunakan salah satu opsi berikut untuk pengaturan ini.

- AllTags(string) — Tambahkan semua tag instance ke wallpaper Anda.

```
instanceTags: AllTags
```

- instanceTags (daftar string) - Tentukan daftar nama tanda instans untuk ditambahkan ke wallpaper Anda. Contoh:

```
instanceTags:
  - Tag 1
  - Tag 2
```

Contoh

Contoh berikut menunjukkan input konfigurasi wallpaper yang mengatur jalur file untuk gambar latar belakang wallpaper, bersama dengan tanda instans bernama `Tag 1` dan `Tag 2`, serta atribut yang menyertakan nama host, ID instans, dan alamat IP privat serta publik untuk instans tersebut.

```
task: setWallpaper
inputs:
  path: C:\ProgramData\Amazon\EC2Launch\wallpaper\Ec2Wallpaper.jpg
  attributes:
  - hostName
  - instanceId
```

```
  - privateIpAddress
  - publicIpAddress
  instanceTags:
  - Tag 1
  - Tag 2
```

> **Note**
>
> Anda harus mengaktifkan tanda dalam metadata untuk menampilkan tanda pada wallpaper. Untuk informasi selengkapnya tentang tanda instans dan metadata, lihat Melihat tag untuk instans EC2 menggunakan metadata instans.

startSsm

Memulai layanan Systems Manager (SSM) setelah Sysprep.

Frekuensi - selalu

AllowedStages — `[PostReady, UserData]`

Masukan - tidak ada

Contoh

```
task: startSsm
```

sysprep

Mereset status layanan, update `unattend.xml`, menonaktifkan RDP, dan menjalankan Sysprep. Tugas ini berjalan hanya setelah semua tugas lainnya selesai.

Frekuensi - sekali

AllowedStages — `[UserData]`

Masukan —

`clean`: (boolean) membersihkan log instans sebelum menjalankan Sysprep

`shutdown`: (boolean) menutup instans setelah menjalankan Sysprep

Contoh

```
task: sysprep
inputs:
clean: true
shutdown: true
```

writeFile

Menuliskan file ke tujuan.

Frekuensi - lihat Input

AllowedStages — `[PostReady, UserData]`

Masukan —

`frequency`: (string) salah satu `once` atau `always`

`destination`: (string) jalur tempat menulis konten

`content`: (string) teks untuk ditulis ke tujuan

Contoh

```
task: writeFile
inputs:
  - frequency: once
  destination: C:\Users\Administrator\Desktop\booted.txt
  content: Windows Has Booted
```

Memecahkan masalah dengan agen EC2launch v2

Bagian ini menunjukkan skenario pemecahan masalah umum untuk EC2Launch v2, informasi tentang menampilkan log peristiwa Windows, serta output dan pesan log konsol.

Topik pemecahan masalah

- Skenario pemecahan masalah umum
- Log peristiwa Windows
- Output log konsol EC2Launch v2

## Skenario pemecahan masalah umum

Bagian ini menunjukkan skenario pemecahan masalah umum dan langkah-langkah penyelesaiannya.

### Skenario

- Layanan gagal menyetel wallpaper
- Layanan gagal menjalankan data pengguna
- Layanan menjalankan tugas hanya satu kali
- Layanan gagal menjalankan tugas
- Layanan menjalankan data pengguna lebih dari satu kali
- Tugas terjadwal dari EC2Launch v1 gagal untuk berjalan setelah migrasi ke EC2Launch v2
- Layanan menginisialisasi volume EBS yang tidak kosong
- Tugas setWallpaper tidak diaktifkan tetapi wallpaper diatur ulang saat reboot
- Layanan macet dalam status berjalan
- agent-config.yml tidak valid mencegah pembukaan kotak dialog pengaturan EC2launch v2
- Task:ExecuteScript harus unik dan hanya dipanggil sekali

### Layanan gagal menyetel wallpaper

#### Resolusi

1. Periksa apakah `%AppData%\Roaming\Microsoft\Windows\Start Menu\Programs \Startup\setwallpaper.lnk` ada.
2. Periksa `%ProgramData%\Amazon\EC2Launch\log\agent.log` untuk melihat apakah ada kesalahan yang terjadi.

### Layanan gagal menjalankan data pengguna

Penyebab potensial: Layanan mungkin gagal sebelum menjalankan data pengguna.

#### Resolusi

1. Periksa `%ProgramData%\Amazon\EC2Launch\state\previous-state.json`.
2. Lihat jika `boot`, `network`, `preReady`, dan `postReadyLocalData` semuanya telah ditandai sebagai sukses.

3. Jika salah satu tahapan gagal, periksa `%ProgramData%\Amazon\EC2Launch\log\agent.log` jika ada kesalahan tertentu.

Layanan menjalankan tugas hanya satu kali

Resolusi

1. Periksa frekuensi tugas.
2. Jika layanan sudah berjalan setelah Sysprep, dan frekuensi tugas diatur ke `once`, tugas tidak akan dijalankan lagi.
3. Atur frekuensi tugas ke `always` jika Anda ingin menjalankan tugas setiap saat EC2Launch v2 berjalan.

Layanan gagal menjalankan tugas

Resolusi

1. Periksa entri terbaru di `%ProgramData%\Amazon\EC2Launch\log\agent.log`.
2. Jika tidak ada kesalahan yang terjadi, coba jalankan layanan secara manual dari `"%ProgramFiles%\Amazon\EC2Launch\EC2Launch.exe" run` untuk melihat apakah tugas berhasil.

Layanan menjalankan data pengguna lebih dari satu kali

Resolusi

Data pengguna ditangani secara berbeda antara EC2Launch v1 dan EC2Launch v2. EC2Launch v1 menjalankan data pengguna sebagai tugas terjadwal pada instans jika `persist` diatur ke `true`. Jika `persist` diatur ke, `false`maka tugas tidak dijadwalkan bahkan saat keluar dengan reboot atau terganggu saat berjalan.

EC2Launch v2 menjalankan data pengguna sebagai tugas agen dan melacak status berjalannya. Jika masalah data pengguna me-restart komputer atau jika data pengguna terganggu saat berjalan, maka status berjalan-nya masih tetap ada sebagai `pending` dan data pengguna akan berjalan lagi pada boot instans berikutnya. Jika Anda ingin mencegah skrip data pengguna berjalan lebih dari sekali, buat skrip menjadi idempoten.

Contoh skrip idempotensi berikut mengatur nama komputer dan bergabung dengan domain.

```
<powershell>
  $name = $env:computername
  if ($name -ne $desiredName) {
 Rename-Computer -NewName $desiredName
  }
  $domain = Get-ADDomain
  if ($domain -ne $desiredDomain)
  {
 Add-Computer -DomainName $desiredDomain
  }
  $telnet = Get-WindowsFeature -Name Telnet-Client
  if (-not $telnet.Installed)
  {
 Install-WindowsFeature -Name "Telnet-Client"
  }
</powershell>
<persist>false</persist>
```

Tugas terjadwal dari EC2Launch v1 gagal untuk berjalan setelah migrasi ke EC2Launch v2

Resolusi

Alat migrasi tidak mendeteksi tugas terjadwal terkait dengan skrip EC2Launch v1; oleh karena itu, tidak secara otomatis mengatur tugas-tugas tersebut di EC2Launch v2. Untuk mengonfigurasi tugas-tugas ini, edit file `agent-config.yml`, atau gunakan kotak dialog pengaturan EC2Launch v2. Sebagai contoh, jika sebuah instans memiliki tugas terjadwal yang menjalankan `InitializeDisks.ps1`, kemudian setelah Anda menjalankan alat migrasi, maka Anda harus menentukan volume yang ingin Anda inisialisasi di kotak dialog pengaturan EC2Launch v2. Lihat Langkah 6 prosedur untuk Ubah pengaturan menggunakan kotak dialog pengaturan EC2Launch v2.

Layanan menginisialisasi volume EBS yang tidak kosong

Resolusi

Sebelum menginisialisasi volume, EC2Launch v2 mencoba untuk mendeteksi apakah volume tersebut kosong. Jika volume tidak kosong, maka ia melewatkan inisialisasi. Setiap volume yang terdeteksi sebagai tidak kosong tidak akan diinisialisasi. Volume dianggap kosong jika 4 KiB pertama volume kosong, atau jika volume tidak memiliki tata letak Windows-recognizable drive. Volume yang diinisialisasi dan diformat pada sistem Linux tidak memiliki tata letak Windows-recognizable drive, misalnya MBR atau GPT. Oleh karena itu, volume tersebut akan dianggap sebagai volume kosong dan diinisialisasi. Jika Anda ingin menyimpan data ini, jangan mengandalkan deteksi hard

disk kosong EC2Launch v2. Sebaliknya, tentukan volume yang ingin Anda inisialisasi di kotak dialog pengaturan EC2Launch v2 (lihat langkah 6) atau dalam `agent-config.yml`.

Tugas setWallpaper tidak diaktifkan tetapi wallpaper diatur ulang saat reboot

Tu `setWallpaper` gas membuat file `setwallpaper.lnk` pintasan di folder startup dari setiap pengguna yang ada. File pintasan ini berjalan saat pengguna masuk untuk pertama kalinya setelah boot instans. File ini menyiapkan instans dengan wallpaper kustom yang menampilkan atribut instans. Menghapus `setWallpaper` tugas tidak menghapus file pintasan ini. Anda harus menghapus file ini secara manual atau menghapusnya menggunakan skrip.

Jalur pintasnya adalah:

`$env:SystemDrive/Users/<user>/AppData/Roaming/Microsoft/Windows/Start Menu/Programs/Startup/setwallpaper.lnk`

Resolusi

Hapus file ini secara manual, atau hapus menggunakan skrip.

Contoh PowerShell skrip untuk menghapus file pintasan

```
foreach ($userDir in (Get-ChildItem "C:\Users" -Force -Directory).FullName)
{
 $startupPath = Join-Path $userDir -ChildPath "AppData\Roaming\Microsoft\Windows\Start
 Menu\Programs\Startup"
 if (Test-Path $startupPath)
 {
  $wallpaperSetupPath = Join-Path $startupPath -ChildPath "setwallpaper.lnk"
  if (Test-Path $wallpaperSetupPath)
  {
   Remove-Item $wallpaperSetupPath -Force -Confirm:$false
  }
 }
}
```

Layanan macet dalam status berjalan

Deskripsi

EC2launch v2 diblokir, dengan pesan log (`agent.log`) mirip dengan berikut ini:

```
2022-02-24 08:08:58 Info:
 ********************************************************************************
2022-02-24 08:08:58 Info: EC2Launch Service starting
2022-02-24 08:08:58 Info: Windows event custom log exists: Amazon EC2Launch
2022-02-24 08:08:58 Info: ACPI SPCR table not supported. Bailing Out
2022-02-24 08:08:58 Info: Serial port is in use. Waiting for Serial Port...
2022-02-24 08:09:00 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:02 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:04 Info: ACPI SPCR table not supported. Use default console port.
2022-02-24 08:09:06 Info: ACPI SPCR table not supported. Use default console port.
```

Kemungkinan penyebab

SAC diaktifkan dan menggunakan port serial. Untuk informasi selengkapnya, lihat Gunakan SAC untuk memecahkan masalah instans Windows Anda.

Resolusi

Coba langkah-langkah berikut untuk mengatasi masalah ini:

- Nonaktifkan layanan yang menggunakan port serial.
- Jika Anda ingin layanan terus menggunakan port serial, tulis skrip khusus untuk melakukan tugas agen peluncuran dan menginvokasinya sebagai tugas terjadwal.

### **`agent-config.yml tidak valid`** mencegah pembukaan kotak dialog pengaturan EC2launch v2

Deskripsi

Pengaturan EC2launch v2 mencoba mengurai file `agent-config.yml` sebelum membuka kotak dialog. Jika file konfigurasi YAMB tidak mengikuti skema yang didukung, kotak dialog akan menampilkan kesalahan berikut:

`Unable to parse configuration file agent-config.yml. Review configuration file. Exiting application.`

Resolusi

1. Verifikasi bahwa file konfigurasi mengikuti skema yang didukung.
2. Jika Anda ingin memulai dari awal, salin file konfigurasi default ke `agent-config.yml`. Anda dapat menggunakan contoh `agent-config.yml` yang disediakan di bagian Konfigurasi Tugas.

3. Anda juga dapat memulai dari awal dengan menghapus `agent-config.yml`. Pengaturan EC2Launch v2 menghasilkan file konfigurasi kosong.

### `Task:ExecuteScript harus unik dan hanya dipanggil sekali`

Deskripsi

Tugas tidak dapat diulang dalam tahap yang sama.

Resolusi

Beberapa tugas harus dimasukkan sebagai array, seperti executeScript dan executeProgram. Untuk contoh cara menulis skrip sebagai array, lihat executeScript.

Log peristiwa Windows

EC2Launch v2 menerbitkan log peristiwa Windows untuk peristiwa penting, seperti layanan dimulai, Windows siap, serta keberhasilan dan kegagalan tugas. Pengidentifikasi peristiwa secara unik mengidentifikasi peristiwa tertentu. Setiap acara berisi informasi tahapan, tugas, dan level, serta deskripsi. Anda dapat mengatur pemicu untuk peristiwa tertentu menggunakan pengenal peristiwa.

ID peristiwa memberikan informasi tentang suatu peristiwa dan mengidentifikasi beberapa peristiwa secara unik. Digit paling signifikan dari ID peristiwa menunjukkan tingkat keparahan suatu peristiwa.

| Peristiwa | Digit paling tidak signifikan |
|---|---|
| `Success` | `. . .0` |
| `Informational` | `. . .1` |
| `Warning` | `. . .2` |
| `Error` | `. . .3` |

Service-related peristiwa yang dihasilkan saat layanan dimulai atau berhenti menyertakan pengidentifikasi peristiwa satu digit.

| Peristiwa | Pengenal digit tunggal |
|---|---|
| `Success` | `0` |

| Peristiwa | Pengenal digit tunggal |
| --- | --- |
| `Informational` | `1` |
| `Warning` | `2` |
| `Error` | `3` |

Pesan peristiwa untuk peristiwa `EC2LaunchService.exe` dimulai dengan `Service:`. Pesan peristiwa untuk peristiwa `EC2Launch.exe` tidak dimulai dengan `Service:`.

Empat digit ID peristiwa mencakup informasi tentang tahap, tugas, dan tingkat keparahan suatu peristiwa.

Topik

- Format ID Peristiwa
- Contoh ID Peristiwa
- Skema log peristiwa Windows

Format ID Peristiwa

Tabel berikut menunjukkan format pengenal peristiwa EC2Launch v2.

| 3 | 2 1 | 0 |
| --- | --- | --- |
| D | T | L |

Huruf dan angka dalam tabel mewakili tipe dan definisi peristiwa berikut.

| Tipe peristiwa | Definisi |
| --- | --- |
| S (Panggung) | 0 - Service-level pesan<br>1 - Boot<br>2 - Jaringan |

| Tipe peristiwa | Definisi |
| --- | --- |
| | 3 - PreReady<br>5 - Windows sudah Siap<br>6 - PostReady<br>7 - Data Pengguna |
| T (Tugas) | Tugas yang diwakili oleh dua nilai yang sesuai berbeda untuk setiap tahap. Untuk melihat daftar lengkap peristiwa, lihat Skema log Peristiwa Windows. |
| L (Level peristiwa) | 0 - Sukses<br>1 - Informasi<br>2 - Peringatan<br>3 - Kesalahan |

Contoh ID Peristiwa

Berikut adalah contoh ID peristiwa.

- `5000` - Windows siap digunakan
- `3010`- Aktifkan tugas windows dalam PreReady tahap berhasil
- `6013`- Mengatur tugas wallpaper di tahap Data PostReady Lokal mengalami kesalahan

Skema log peristiwa Windows

| MessageId/Event Id | Pesan peristiwa |
| --- | --- |
| `. . .0` | `Success` |

| MessageId/Event Id | Pesan peristiwa |
| --- | --- |
| . . .1 | Informational |
| . . .2 | Warning |
| . . .3 | Error |
| x | EC2Launch service-level logs |
| 0 | EC2Launch service exited successfully |
| 1 | EC2Launch service informational logs |
| 2 | EC2Launch service warning logs |
| 3 | EC2Launch service error logs |
| 10 | Replace state.json with previous-state.json |
| 100 | Serial Port |
| 200 | Sysprep |
| 300 | PrimaryNic |
| 400 | Metadata |
| x000 | Stage (1 digit), Task (2 digits), Status (1 digit) |
| 1000 | Boot |
| 1010 | Boot - extend_root_partition |
| 2000 | Network |
| 2010 | Network - add_routes |

| MessageId/Event Id | Pesan peristiwa |
| --- | --- |
| 3000 | PreReady |
| 3010 | PreReady - activate_windows |
| 3020 | PreReady - install_egpu_manager |
| 3030 | PreReady - set_monitor_on |
| 3040 | PreReady - set_hibernation |
| 3050 | PreReady - set_admin_account |
| 3060 | PreReady - set_dns_suffix |
| 3070 | PreReady - set_wallpaper |
| 3080 | PreReady - set_update_schedule |
| 3090 | PreReady - output_log |
| 3100 | PreReady - enable_open_ssh |
| 5000 | Windows is Ready to use |
| 6000 | PostReadyLocalData |
| 7000 | PostReadyUserData |
| 6010/7010 | PostReadyLocal/UserData - set_wallpaper |
| 6020/7020 | PostReadyLocal/UserData - set_update_schedule |
| 6030/7030 | PostReadyLocal/UserData - set_hostname |
| 6040/7040 | PostReadyLocal/UserData - execute_program |

| MessageId/Event Id | Pesan peristiwa |
| --- | --- |
| 6050/7050 | PostReadyLocal/UserData - execute_script |
| 6060/7060 | PostReadyLocal/UserData - manage_package |
| 6070/7070 | PostReadyLocal/UserData - initialize_volume |
| 6080/7080 | PostReadyLocal/UserData - write_file |
| 6090/7090 | PostReadyLocal/UserData - start_ssm |
| 7100 | PostReadyUserData - enable_open_ssh |
| 6110/7110 | PostReadyLocal/UserData - enable_jumbo_frames |

### Output log konsol EC2Launch v2

Bagian ini berisi contoh output log konsol untuk EC2Launch v2 dan menampilkan daftar semua pesan kesalahan log konsol EC2Launch v2 untuk membantu Anda memecahkan masalah. Untuk informasi selengkapnya tentang output konsol instance dan cara mengaksesnya, lihatthe section called “Output konsol instans”.

Output

- Output log konsol EC2Launch v2
- Pesan log konsol EC2Launch v2

### Output log konsol EC2Launch v2

Berikut ini adalah contoh output log konsol untuk EC2Launch v2. Beberapa nilai dalam contoh ini diganti dengan teks representatif yang dikelilingi oleh kurung kurung keriting.

```
2025/07/22 21:26:53Z: Windows sysprep configuration complete.
2025/07/22 21:26:53Z: Message: Waiting for access to metadata...
2025/07/22 21:26:53Z: Message: Meta-data is now available.
2025/07/22 21:26:53Z: AMI Origin Version: 2024.12.13
2025/07/22 21:26:53Z: AMI Origin Name: Windows_Server-2022-English-Full-Base
2025/07/22 21:26:53Z: OS: Microsoft Windows NT 10.0.20348
2025/07/22 21:26:53Z: OsVersion: 10.0
2025/07/22 21:26:53Z: OsProductName: Windows Server 2022 Datacenter
2025/07/22 21:26:53Z: OsBuildLabEx: 20348.1.amd64fre.fe_release.210507-1500
2025/07/22 21:26:53Z: OsCurrentBuild: 20348
2025/07/22 21:26:53Z: OsReleaseId: 2009
2025/07/22 21:26:53Z: Language: en-US
2025/07/22 21:26:53Z: TimeZone: UTC
2025/07/22 21:26:53Z: Offset: UTC +0000
2025/07/22 21:26:53Z: Launch: EC2 Launch v2.2.63
2025/07/22 21:26:53Z: AMI-ID: ami-1234567890abcdef1
2025/07/22 21:26:53Z: Instance-ID: i-1234567890abcdef0
2025/07/22 21:26:54Z: Instance Type: t3.xlarge
2025/07/22 21:26:54Z: Driver: AWS NVMe Driver v1.6.0.35
2025/07/22 21:26:54Z: SubComponent: 1.6.0.35; EnableSCSIPersistentReservations: 0
2025/07/22 21:26:54Z: Driver: AWS PV Driver Package v8.5.0
2025/07/22 21:26:55Z: Driver: Amazon Elastic Network Adapter v2.8.0.0
2025/07/22 21:26:55Z: Driver: EC2 Windows Utility Device v3.1.0.25
2025/07/22 21:26:55Z: HOSTNAME: EC2AMAZ-9FJG5CC
2025/07/22 21:26:55Z: RDPCERTIFICATE-SUBJECTNAME: {certificate subject name}
2025/07/22 21:26:55Z: RDPCERTIFICATE-THUMBPRINT: {thumbprint hash}
2025/07/22 21:26:56Z: SSM: Amazon SSM Agent v3.3.2746.0
2025/07/22 21:26:57Z: User data format: no_user_data
2025/07/22 21:26:57Z: EC2LaunchTelemetry: IsTelemetryEnabled=true
2025/07/22 21:26:57Z: EC2LaunchTelemetry: AgentOsArch=windows_amd64
2025/07/22 21:26:57Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2025/07/22 21:26:57Z: EC2LaunchTelemetry: AgentCommandErrorCode=1
2025/07/22 21:26:57Z: EC2LaunchTelemetry: AdminPasswordTypeCode=0
2025/07/22 21:26:57Z: EC2LaunchTelemetry: AgentErrorLocation=execute_windows.go:410
2025/07/22 21:26:57Z: EC2LaunchTelemetry: IpConflictDetectionCode=0
2025/07/22 21:26:57Z: Message: Windows is Ready to use
{"type":"EC2AgentTelemetry","agentId":"WindowsLaunchAgentV2", ...}
{"type":"EC2AgentTelemetry","agentId":"WindowsLaunchAgentV2", ...}
```

### Pesan log konsol EC2Launch v2

Berikut ini adalah daftar semua pesan log konsol EC2Launch v2.

```
Error EC2Launch service is stopping. {error message}
```

Detail kesalahan layanan yang dihentikan:

- `Error setting up EC2Launch agent folders`
- `See instance logs for detail`
- `Error stopping service`
- `Error initializing service`

```
Windows sysprep configuration complete
```

```
Invalid administrator username: {invalid username}
```

```
Invalid administrator password
Username: {username}
Password: <Password>{encrypted password}</Password>
```

Pesan berikut adalah blok informasi yang berisi detail AMI:

```
AMI Origin Version: {amiVersion}
AMI Origin Name: {amiName}
Microsoft Windows NT {currentVersion}.{currentBuildNumber}
OsVersion: {currentVersion}
OsProductName: {productName}
OsBuildLabEx: {buildLabEx}
OsCurrentBuild: {currentBuild}
OsReleaseId: {releaseId}
Language: {language}
TimeZone: {timeZone}
Offset: UTC {offset}
Launch agent: EC2Launch {BuildVersion}
AMI-ID: {amiId}
Instance-ID: {instanceId}
Instance Type: {instanceType}
HOSTNAME: {computer name}
RDPCERTIFICATE-SUBJECTNAME: {certificate subject name}
RDPCERTIFICATE-THUMBPRINT: {thumbprint hash}
SqlServerBilling: {sql billing}
```

```
SqlServerInstall: {sql patch leve, edition type}
Driver: AWS NVMe Driver {version}
Driver: Inbox NVMe Driver {version}
Driver: AWS PV Driver Package {version}
Driver: EC2 Windows Utility Device {version}
SSM: Amazon SSM Agent {version}
AWS VSS Version: {version}
```

```
Windows sysprep configuration complete.
Windows is being configured. 'SysprepState is {state}'
Windows is still being configured. 'SysprepState is {state}'
Windows is Ready to use
Waiting for access to metadata...
Meta-data is now available.
Metadata is not available for this instance.
Timed out waiting for access to metadata.
User data format: {format}
```

Pesan telemetri EC2launch v2 menyertakan nilai properti telemetri peluncuran. Dimulai dengan versi 2.2.63, data telemetri agen EC2 diformat sebagai objek JSON.

```
EC2LaunchTelemetry: {telemetry property}
```

```
{"type":"EC2AgentTelemetry","agentId":"WindowsLaunchAgentV2" ... }
```

Riwayat versi EC2Launch v2

Riwayat versi

- Riwayat versi EC2Launch v2

Riwayat versi EC2Launch v2

Untuk memastikan bahwa Anda telah menginstal agen peluncuran terbaru, lihatInstal EC2Launch v2 versi terbaru. Anda dapat menerima pemberitahuan saat versi baru agen EC2launch v2 dirilis. Untuk informasi selengkapnya, lihat Berlangganan pemberitahuan agen peluncuran EC2 Windows.

Tabel berikut menjelaskan versi EC2Launch v2 yang dirilis.

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.5.2 | • Memperbaiki masalah saat validasi sertifikat RDP gagal untuk sertifikat kustom dan Policy-deployed sertifikat Grup. | Agustus 5, 2026 |
| 2.5.1 | • Peningkatan deteksi kesalahan saat mengaktifkan hibernasi. | 3 Juni 2026 |
| 2.5.0 | • Diperbarui logika port serial untuk melanjutkan tugas agen ketika port serial tidak tersedia.<br>• Menambahkan versi driver Perangkat Utilitas EC2 Windows ke output konsol instance.<br>• Menghapus file bench.log yang tidak digunakan.<br>• Pencarian driver yang dioptimalkan saat menulis ke output konsol instance. | 4 Mei 2026 |
| 2.4.0 | • Pencatatan agen yang disederhanakan dengan menghapus `err.log` file yang berlebihan`console.log` .<br>• Kemampuan pencatatan agen ditingkatkan dengan menambahkan rotasi log, membatasi ukuran log hingga 1 MB dan meningkatkan presisi waktu hingga milidetik.<br>• Mengh Hyper-V apus kueri status dan output konsol instance terkait. | Maret 5, 2026 |
| 2.3.237 | • Memperbaiki masalah kerusakan agen yang disebabkan oleh ketidakcocokan antara versi runtime Visual C++ tertentu dan fungsi telemetri agen. | Februari 25, 2026 |
| 2.3.108 | • Menambahkan tampilan alamat IPv6 ke wallpaper. Untuk atribut Memori dan Jaringan, lihat AWS Dokumentasi. | Desember 29, 2025 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.3.56 | • Memperbaiki masalah di mana rute jaringan persisten yang telah dikonfigurasi sebelumnya tidak ditangani dengan benar oleh EC2launch v2. | 4 November 2025 |
| 2.3.5 | • Peningkatan deteksi antarmuka jaringan utama menggunakan IMDS saat menambahkan rute. | 15. September 2025 |
| 2.2.63 | • Agen peluncuran yang diperbarui untuk meningkat kanTelemetri. | Juli 29, 2025 |
| 2.1.1 | • Menambahkan dukungan penuh untuk IPv6. | Mei 14, 2025 |
| 2.0.2107 | • Penambahan rute yang ditingkatkan untuk menangani skenario ketika alamat IPv4 atau IPv6 tidak tersedia di antarmuka. | Maret 27, 2025 |
| 2.0.2081 | • Memperbaiki masalah di mana informasi sertifikat RDP tidak diambil atau divalidasi dengan benar. Menambahk an fungsionalitas untuk secara otomatis memulai Layanan Desktop Jarak Jauh jika diperlukan.<br>• Izin layanan EC2launch v2 yang disesuaikan untuk memperbaiki masalah yang terjadi saat menanyakan status layanan. | Februari 4, 2025 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.2046 | • Memperbarui jalur wallpaper dalam `agent-config.yml` file untuk menggunakan jalur wallpaper sistem operasi default.<br>• Menambahkan telemetri untuk memantau lokasi di mana kesalahan agen terjadi.<br>• Pesan log agen yang diperbarui. | 3 Oktober 2024 |
| 2.0.1981 | • Pesan kesalahan perintah `EC2Launch.exe` CLI yang diperbarui untuk pengguna non-Administrator. | Agustus 6, 2024 |
| 2.0.1948 | • Menambahkan telemetri untuk memantau penggunaan opsi kata sandi admin.<br>• Izin EC2launch yang dimodifikasi. | Juli 1, 2024 |
| 2.0.1924 | • Memperbarui UI Pengaturan EC2Launch.<br>• Memperbarui perintah wallpaper CLI.<br>• Memperbarui penginstal EC2launch. | Juni 10, 2024 |
| 2.0.1914 | • Tambahkan rute dengan alamat gateway yang tidak ditentukan (`0.0.0.0`untuk IPv4 atau `::` untuk IPv6).<br>• Selalu tambahkan rute IPv4 dan IPv6.<br>• Memperbaiki masalah ketika `Administrator` nama pengguna ditambahkan ke `agent-config.yml` file saat tidak ditentukan.<br>• Izin EC2launch v2 yang dimodifikasi. | 5 Juni 2024 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1881 | • Menambahkan opsi kata sandi terenkripsi untuk `setAdminAccount` tugas.<br>• Menambahkan perintah CLI untuk mengenkripsi kata sandi statis di agent-config.yml.<br>• Memperbaiki masalah di mana data pengguna XML tidak menambahkan PowerShell argumen saat dijalankan dengan izin Administrator. Untuk detail selengkapnya, lihat Cara Amazon EC2 menangani data pengguna untuk instans Windows.<br>• PowerShell Argumen yang disesuaikan untuk `executeScript` tugas dan skrip data pengguna saat dijalankan dengan `LocalSystem` izin. Ketika argumen kosong, agen menggunakan nilai default berikut:`-ExecutionPolicy Unrestricted` .<br>• Mencegah pencetakan versi driver duplikat ke log konsol. | 8 Mei 2024 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1815 | • Penanganan kesalahan yang disesuaikan agar gagal pada masalah penyiapan kritis sebelum sysprep.<br>• Memperbaiki masalah di mana tugas wallpaper dan nama host dapat menggunakan alamat IP yang salah pada instance dengan beberapa alamat IP yang ditetapkan ke antarmuka jaringan utama.<br>• Wallpaper dan tugas nama host diubah untuk mendapatkan IP pribadi dari IMDS terlebih dahulu, kemudian gagal kembali ke WMI jika IMDS dinonaktifkan.<br>• Memperbaiki masalah dengan `initializeVolume` tugas di mana `sc1` volume gagal diinisialisasi karena kesalahan sementara. | Maret 6, 2024 |
| 2.0.1739 | • Memperbaiki masalah yang mencegah kode keluar ditangkap oleh `executeScript` tugas yang dijalankan sebagai pengguna Administrator Windows. | Januari 17, 2024 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1702 | • Membatasi izin `Telemetry.log` menjadi `read-execute` saja untuk pengguna standar.<br>• Mengonfigurasi layanan EC2Launch Windows untuk memulai kembali pada kegagalan start-up.<br>• Membuat kegagalan `add-routes` dapat ditindaklanjuti dengan melakukan logging output `route.exe stderr`.<br>• Memperbaiki masalah yang terjadi saat metrik rute berada di luar jangkauan [1, 9999].<br>• Menambahkan dukungan wallpaper ke beberapa tipe instans baru.<br>• Memperbaiki masalah yang disebabkan oleh skrip data pengguna yang berjalan sebagai pengguna Administrator Windows dan mengirim output ke `stderr`. | 4 Januari 2024 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1643 | • Memperbarui alat `ebsnvme-id.exe` ke versi 1.1.0.7.<br>• Memperbaiki masalah dengan menerima penskalaan sisi (RSS) dan menerima pengaturan kedalaman antrean pada tipe instans logam yang dimulai dengan 'logam-*', seperti logam-48x1.<br>• Peristiwa telemetri yang dihapus yang melaporkan perintah data pengguna XML yang memblokir agen.<br>• Tugas `setDnsSuffix` yang diperbarui untuk membatasi devolusi nama domain berdasarkan entri registri: `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel` .<br>• Menambahkan tugas publik dan CLI yang menambahkan rute jaringan.<br>• Catatan — Ini adalah versi terakhir yang secara resmi mendukung Windows Server 2012.<br>• Catatan — Ini adalah versi terakhir yang secara resmi mendukung sistem operasi 32-bit. | 4 Oktober 2023 |
| 2.0.1580 | • Cara agen peluncuran menangani kesalahan saat Anda mengubah izin file log diubah.<br>• Menambahkan batas waktu habis untuk menghubungkan ke port serial. Batas waktu memungkinkan agen peluncuran untuk terus berjalan jika port serial sedang digunakan. | 5 September 2023 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1521 | • Bendera –block usang dari EC2Launch.exe reset dan perintah sysprep.<br>• EC2Launch.exe yang diperbarui untuk mendeteksi dan menangani perintah reset dan sysprep yang digunakan dalam tugas executeScript inline. Perintah tersebut menyebabkan agen berhenti berjalan setelah executeScript tugas menjalankannya.<br>• Skrip data pengguna XML yang diperbarui untuk menjalankan inline secara default.<br>• Aktifkan skrip data pengguna XML untuk berjalan terpisah dengan tanda baru detach. Untuk detail selengkapnya, lihat Skrip data pengguna.<br>• Membuat perubahan berikut ini pada log agen.<br>  • Pesan log agen yang diperbarui.<br>  • Menghapus executeScript konten dan output dari log agen.<br>  • Argumen executeProgram dan output yang dihapus dari log agen.<br>• Membuat perubahan berikut pada log konsol.<br>  • Menambahkan nilai EnableSCSIPersistentReservations ke log konsol. | 3 Juli 2023 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1303 | • Menambahkan penanganan kesalahan tambahan dan baris log saat menambahkan rute jaringan.<br>• Diizinkan `executeScript` dan `executeProgram` tugas di pang PreReady gung.<br>• Tugas `executeProgram` yang diperbarui untuk menghasil kan file output yang mirip dengan output dari tugas executeSc ript. Untuk informasi selengkapnya, lihat executeProgram.<br>• Menambahkan telemetri untuk memantau penggunaan perintah agen pemblokiran dalam data pengguna XML. | 3 Mei 2023 |
| 2.0.1245 | • Peningkatan visibilitas tentang kerusakan dengan mencatat tumpukan panggilan rusak dalam teks yang jelas.<br>• Menambahkan EventLog layanan sebagai dependensi startup untuk memperbaiki kerusakan saat layanan Amazon EC2launch dimulai lebih cepat daripada layanan. EventLog<br>• Membuat data pengguna XML dijalankan sebelum PostReady tahap dari file konfigurasi agen (seperti EC2launch v1 dan EC2config).<br>• Menambahkan data pengguna YAML versi 1.1 untuk membuat data pengguna berjalan sebelum PostReady tahap dari file konfigurasi agen (data pengguna YAML versi 1.0 berjalan setelah PostReady tahap dari file konfigurasi agen). | 8 Maret 2023 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1173 | • Menambahkan fitur opsional untuk menampilkan tanda instans pada wallpaper. Untuk informasi selengkapnya, lihat setWallpaper.<br>• Menambahkan penanganan kesalahan saat grup keamanan untuk Elastic Graphics tidak disiapkan dengan benar.<br>• Memperbaiki batas waktu saat Layanan Metadata Instans tidak diaktifkan. | 6 Februari 2023 |
| 2.0.11121 | • Memperbaiki masalah saat kesalahan 404 dicetak ke wallpaper saat tidak ada alamat IPv4 publik yang ditetapkan.<br>• Memperbaiki masalah saat sistem file volume diformat sebagai `RAW` ganti `NTFS` saat huruf drive perangkatnya disetel ke. `D`<br>• Memperbaiki masalah di mana volume SSD NVMe salah diidentifikasi sebagai volume EBS.<br>• Memperbaiki kesalahan saat mengaktifkan Windows saat IMDS dinonaktifkan. | 4 Januari 2023 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.1082 | • Memperbaiki masalah di mana bidang `setWallpaper : privateIpAddress` kosong saat IMDS dinonaktifkan.<br>• Memperbaiki masalah dengan mengatur nama host ke alamat IPv4 pribadi saat IMDS dinonaktifkan.<br>• Memperbaiki masalah dengan menginisialisasi volume pada Windows Server 2012.<br>• Memperbaiki masalah dengan pengaturan bingkai jumbo.<br>• Memperbaiki kesalahan ketika tidak ada kunci SSH yang ditentukan pada peluncuran instans.<br>• Memperbaiki kesalahan pada Windows Server 2012 ketika Windows tidak memiliki kunci registri ReleaseId ''. | 7 Desember 2022 |
| 2.0.1011 | • Memperbaiki logika untuk menemukan adaptor jaringan ketika PnPDeviceID kosong. | 11 November 2022 |
| 2.0.1009 | • Menggunakan informasi segmen PCI untuk memilih port konsol. | 8 November 2022 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.982 | • Menambahkan logika coba lagi untuk mendapatkan informasi RDP.<br>• Memperbaiki kesalahan selama inisialisasi volume pada `d2.8xlarge` instans.<br>• Memperbaiki masalah di mana adaptor jaringan yang salah dapat dipilih setelah reboot.<br>• Menghapus pesan kesalahan alarm palsu saat ACPI SPCR tidak tersedia. | 31 Oktober 2022 |
| 2.0.863 | • Logika tunggu IMDS yang diperbarui untuk membuat permintaan IMDSv2 saja.<br>• Menambahkan logika untuk menetapkan huruf drive ke volume yang sudah diinisialisasi tetapi tidak dipasang.<br>• Mencetak pesan kesalahan yang lebih spesifik ketika jenis key pair tidak didukung.<br>• Memperbaiki bug kode reboot 3010.<br>• Menambahkan pemeriksaan untuk data pengguna yang dengan enkode base64 tidak valid. | 6 Juli 2022 |
| 2.0.698 | • Memperbaiki kesalahan ketik dalam output log saat menjalank an skrip. | 30 Januari 2022 |
| 2.0.674 | • Telemetri mengunggah kontrol enabled/disabled privasi.<br>• Memperbaiki `index out of bounds` bug.<br>• Menghapus pintasan wallpaper selama `sysprep`. | 15 November 2021 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.651 | • Menambahkan logika untuk menghapus instalan agen warisan selama instalasi EC2launch v2.<br>• Memperbaiki masalah `list-volume` CLI saat volume root tidak terdaftar sebagai volume 0. | 7 Oktober 2021 |
| 2.0.592 | • Memperbaiki bug untuk melaporkan status tahap dengan benar.<br>• Menghapus pesan kesalahan alarm palsu saat file log ditutup.<br>• Menambahkan telemetri. | 31 Agustus 2021 |
| 2.0.548 | • Menambahkan angka nol di depan untuk nama host IP hex.<br>• Memperbaiki izin file untuk `enableOpenSsh` tugas.<br>• Memperbaiki crash perintah sysprep. | 4 Agustus 2021 |
| 2.0.470 | • Memperbaiki bug di tahap jaringan untuk menunggu DHCP menetapkan IP ke instans.<br>• Memperbaiki bug dengan `setDnsSuffix` ketika kunci `SearchList` registri tidak ada.<br>• Memperbaiki bug dalam logika devolusi DNS di. `setDnsSuffix`<br>• Menambahkan rute jaringan setelah reboot perantara.<br>• Mengizinkan `initializeVolume` untuk menulis ulang volume yang ada.<br>• Menghapus informasi tambahan dari subperintah versi. | 20 Juli 2021 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.285 | • Menambahkan opsi untuk menjalankan skrip pengguna dalam proses terpisah.<br>• Userdata warisan (userdata XML) sekarang berjalan dalam proses terpisah, yang merupakan perilaku yang sama dengan agen peluncuran sebelumnya.<br>• Menambahkan bendera CLI ke perintah `sysprep` dan `reset`, yang memungkinkannya untuk memblokir sampai layanan berhenti.<br>• Membatasi izin folder konfigurasi. | 8 Maret 2021 |
| 2.0.207 | • Menambahkan bidang `hostName` opsional ke tugas `setHostName` .<br>• Memperbaiki bug reboot. Reboot tugas `executeScript` dan `executeProgram` akan ditandai sebagai berjalan.<br>• Menambahkan lebih banyak kode kembali untuk perintah status.<br>• Menambahkan layanan bootstrap untuk memperbaiki masalah startup saat menjalankan tipe instans `t2.nano`.<br>• Memperbaiki mode instalasi bersih untuk menghapus file yang tidak terlacak oleh installer. | 2 Februari 2021 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.160 | • Memperbaiki perintah `validate` untuk mendeteksi nama tahap yang tidak valid.<br>• Menambahkan perintah `w32tm resync` di tugas `addroutes` .<br>• Memperbaiki masalah dengan mengubah urutan pencarian akhiran DNS.<br>• Menambahkan syarat pemeriksaan agar bisa melaporkan data pengguna yang tidak valid dengan lebih baik. | 4 Desember 2020 |
| 2.0.153 | Menambahkan fungsionalitas Sysprep di. UserData | 3 November 2020 |
| 2.0.146 | • Memperbaiki masalah RootExtend pada AMI non-bahasa Inggris.<br>• Memberi izin menulis grup pengguna ke file log.<br>• Membuat partisi MS Reserved untuk volume GPT.<br>• Menambahkan perintah list-volumes dan tarik-turun volume di pengaturan Amazon EC2Launch.<br>• Menambahkan perintah get-agent-config untuk mencetak file agent-config.yml dalam format yaml atau json.<br>• Hapus kata sandi statis jika tidak ada kunci publik yang terdeteksi. | 6 Oktober 2020 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.0.124 | • Menambahkan opsi untuk menampilkan versi OS pada wallpaper.<br>• Menginisialisasi volume EBS yang dienkripsi.<br>• Menambahkan rute untuk VPC tanpa nama DNS lokal. | 10 September 2020 |
| 2.0.104 | • Membuat daftar pencarian sufiks DNS jika tidak ada.<br>• Lewati mode Hibernasi jika tidak diminta. | 12 Agustus 2020 |
| 2.0.0 | Pelepasan awal. | Selasa, Selasa, 30 Juni 2020 |

## Gunakan agen EC2launch v1 untuk melakukan tugas selama peluncuran instance EC2 Windows

AMI yang dikelola Amazon untuk Windows Server 2016 dan 2019 menyertakan satu set skrip Windows Powershell yang disebut EC2launch. EC2Launch melakukan tugas selama boot instans awal. Untuk informasi tentang versi EC2Launch yang disertakan dalam AMI AWS Windows, lihat Referensi AMI AWS Windows.

> **Note**
>
> Agen peluncuran terbaru untuk Windows Server 2016 dan versi sistem operasi yang lebih baru adalah EC2launch v2, yang menggantikan EC2config dan EC2launch, dan hadir pra-instal di AMI AWS Windows Server 2016 dan 2019 dengan nama yang dimulai dengan. `EC2LaunchV2-Windows_Server-*` Anda juga dapat menginstal dan mengkonfigurasi agen secara manual di Windows Server 2016 dan 2019. Untuk informasi selengkapnya, lihat Instal EC2launch v2.
> Untuk menggunakan EC2launch dengan IMDSv2, versinya harus 1.3.2002730 atau setelahnya.

Anda dapat menggunakan PowerShell perintah Windows berikut untuk memverifikasi versi EC2launch yang diinstal.

```
Test-ModuleManifest -Path "C:\ProgramData\Amazon\EC2-Windows\Launch\Module
\Ec2Launch.psd1" | Select Version
```

Tugas EC2Launch

EC2Launch melakukan tugas berikut secara default selama boot instans awal:

- Siapkan wallpaper baru yang menampilkan informasi tentang instans.
- Menetapkan nama komputer ke alamat IPv4 pribadi dari instance.
- Mengirimkan informasi instans ke konsol Amazon EC2.
- Mengirim sidik jari sertifikat RDP ke konsol EC2.
- Mengatur kata sandi acak untuk akun administrator.
- Menambahkan sufiks DNS.
- Secara dinamis memperluas partisi sistem operasi untuk menyertakan ruang yang tidak dipartisi.
- Menjalankan data pengguna (jika ditentukan). Untuk informasi selengkapnya tentang menentukan data pengguna, lihat Jalankan perintah saat Anda meluncurkan instance EC2 dengan input data pengguna.
- Mengatur rute statis persisten untuk menjangkau layanan metadata dan server AWS KMS .

> **Important**
>
> Jika AMI kustom dibuat dari instans ini, rute ini diambil sebagai bagian dari konfigurasi OS dan setiap instans baru yang diluncurkan dari AMI akan mempertahankan rute yang sama, terlepas dari penempatan subnet. Untuk memperbarui rute, lihatPerbarui metadata/KMS rute untuk Server 2016 dan yang lebih baru saat meluncurkan AMI khusus.

Tugas berikut membantu menjaga kompatibilitas ke belakang dengan file layanan EC2Config. Anda juga dapat mengonfigurasi EC2Launch untuk melakukan tugas-tugas ini selama startup:

- Inisialisasi volume EBS sekunder.
- Kirim log Peristiwa Windows ke log konsol EC2.
- Kirim pesan siap digunakan Windows ke konsol EC2.

### Struktur direktori EC2Launch

EC2Launch diinstal secara default di Windows Server 2016 dan AMI setelahnya di direktori root `C:\ProgramData\Amazon\EC2-Windows\Launch`.

> **Note**
>
> Secara default, Windows menyembunyikan file dan folder dalam `C:\ProgramData`. Untuk melihat direktori dan file EC2Launch, Anda harus mengetikkan jalur di Windows Explorer atau ubah properti folder untuk menampilkan file dan folder tersembunyi.

Direktori `Launch` berisi subdirektori berikut.

- `Scripts`- Berisi PowerShell skrip yang membentuk EC2launch.
- `Module` — Berisi modul untuk membangun skrip yang terkait dengan Amazon EC2.
- `Config` — Berisi file konfigurasi skrip yang dapat Anda sesuaikan.
- `Sysprep` — Berisi sumber daya Sysprep.
- `Settings` — Berisi aplikasi untuk antarmuka pengguna grafis Sysprep.
- `Library` — Berisi pustaka bersama untuk agen peluncuran EC2.
- `Log`- Berisi subdirektori untuk skrip dan file log yang dihasilkan oleh skrip.

### Telemetri

Telemetri adalah informasi tambahan yang membantu AWS memahami kebutuhan Anda dengan lebih baik, mendiagnosis masalah, dan memberikan fitur untuk meningkatkan pengalaman Anda dengan AWS layanan.

EC2launch versi `1.3.2003498` dan setelahnya mengumpulkan telemetri, seperti metrik penggunaan dan kesalahan. Data ini dikumpulkan dari instans Amazon EC2 tempat EC2Launch berjalan. Ini termasuk semua AMI Windows yang dimiliki oleh AWS.

Tipe telemetri berikut dikumpulkan oleh EC2launch:

- Informasi penggunaan — perintah agen, metode penginstalan, dan frekuensi eksekusi terjadwal.
- Kesalahan dan informasi diagnostik - instalasi agen dan menjalankan kode kesalahan.

Contoh data yang dikumpulkan:

```
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsAgentScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: IsUserDataScheduledPerBoot=true
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandCode=1
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentCommandErrorCode=5
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallCode=2
2021/07/15 21:44:12Z: EC2LaunchTelemetry: AgentInstallErrorCode=0
```

Telemetri tidak diaktifkan secara default. Anda dapat menonaktifkan kumpulan telemetri kapan saja. Jika telemetri diaktifkan, EC2launch v2 mengirimkan data telemetri tanpa notifikasi pelanggan tambahan.

Pilihan Anda untuk mengaktifkan atau menonaktifkan telemetri dikumpulkan.

Anda dapat memilih masuk atau keluar dari kumpulan telemetri. Pilihan Anda untuk mengikuti atau tidak mengikuti telemetri dikumpulkan untuk memastikan bahwa kami mematuhi opsi telemetri Anda.

Visibilitas telemetri

Saat telemetri diaktifkan, telemetri muncul di output konsol Amazon EC2 sebagai berikut:

```
2021/07/15 21:44:12Z: Telemetry: <Data>
```

Menonaktifkan telemetri pada sebuah instans

Untuk menonaktifkan telemetri dengan menyetel variabel lingkungan sistem, jalankan perintah berikut sebagai administrator:

```
setx /M EC2LAUNCH_TELEMETRY 0
```

Untuk menonaktifkan telemetri selama instalasi, jalankan `install.ps1` sebagai berikut:

```
. .\install.ps1 -EnableTelemetry:$false
```

Topik lainnya untuk EC2launch

- Instal EC2Launch versi terbaru
- Konfigurasikan agen EC2launch v1 pada instans Windows Anda

- Riwayat versi EC2Launch

Instal EC2Launch versi terbaru

Gunakan prosedur berikut untuk mengunduh dan menginstal EC2Launch versi terbaru pada instans Anda.

Untuk mengunduh dan menginstal EC2Launch versi terbaru

1. Jika Anda sudah menginstal dan mengonfigurasi EC2Launch pada sebuah instans, buat cadangan file konfigurasi EC2Launch. Proses penginstalan tidak menyimpan perubahan dalam file ini. Secara default, file terletak di direktori `C:\ProgramData\Amazon\EC2-Windows\Launch\Config`.
2. Unduh EC2-Windows-Launch.zip ke direktori pada instance.
3. Unduh install.ps1 ke direktori yang sama tempat Anda mengunduh `EC2-Windows-Launch.zip`.
4. Jalankan `install.ps1`
5. Jika Anda membuat cadangan file konfigurasi EC2Launch, salin file ke direktori `C:\ProgramData\Amazon\EC2-Windows\Launch\Config`.

Untuk mengunduh dan menginstal versi terbaru dari EC2launch menggunakan PowerShell

Jika Anda sudah menginstal dan mengonfigurasi EC2Launch pada sebuah instans, buat cadangan file konfigurasi EC2Launch. Proses penginstalan tidak menyimpan perubahan dalam file ini. Secara default, file terletak di direktori `C:\ProgramData\Amazon\EC2-Windows\Launch\Config`.

Untuk menginstal versi terbaru EC2launch menggunakan PowerShell, jalankan perintah berikut dari PowerShell jendela sebagai administrator:

```
mkdir $env:USERPROFILE\Desktop\EC2Launch
$Url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Launch/latest/EC2-Windows-Launch.zip"
$DownloadZipFile = "$env:USERPROFILE\Desktop\EC2Launch\" + $(Split-Path -Path $Url -Leaf)
Invoke-WebRequest -Uri $Url -OutFile $DownloadZipFile
$Url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Launch/latest/install.ps1"
$DownloadZipFile = "$env:USERPROFILE\Desktop\EC2Launch\" + $(Split-Path -Path $Url -Leaf)
```

```
Invoke-WebRequest -Uri $Url -OutFile $DownloadZipFile
& $env:USERPROFILE\Desktop\EC2Launch\install.ps1
```

> **Note**
>
> Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:
>
> ```
> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
> ```

Verifikasi instalasi dengan memeriksa agen peluncuran. Jalankan perintah berikut dari PowerShell jendela sebagai administrator:

```
Import-Module C:\ProgramData\Amazon\EC2-Windows\Launch\Module\Ec2Launch.psm1
Import-LocalizedData -BaseDirectory C:\ProgramData\Amazon\EC2-Windows\Launch\Module\ -
FileName 'Ec2Launch.psd1' -BindingVariable moduleManifest
$moduleManifest.Get_Item('ModuleVersion')
```

Konfigurasikan agen EC2launch v1 pada instans Windows Anda

Setelah instans Anda diinisialisasi untuk pertama kalinya, Anda dapat mengonfigurasi EC2Launch untuk berjalan lagi dan melakukan tugas start-up yang berbeda.

Tugas

- Konfigurasikan tugas inisialisasi
- Jadwalkan EC2Launch untuk dijalankan di setiap boot
- Inisialisasi drive dan petakan huruf drive
- Kirim log peristiwa Windows ke log konsol EC2.
- Kirim pesan Windows siap setelah boot berhasil

Konfigurasikan tugas inisialisasi

Tentukan pengaturan di file `LaunchConfig.json` untuk mengaktifkan atau menonaktifkan tugas inisialisasi berikut:

- Atur nama komputer ke alamat IPv4 privat instans.
- Atur monitor agar selalu menyala.
- Siapkan wallpaper baru.
- Tambahkan daftar sufiks DNS.

> **Note**
>
> Ini menambahkan pencarian akhiran DNS untuk domain berikut dan mengonfigurasi akhiran standar lainnya. Untuk informasi selengkapnya tentang cara agen peluncuran mengatur sufiks DNS, lihat. Konfigurasikan Sufiks DNS untuk agen peluncuran EC2 Windows
>
> ```
> region.ec2-utilities.amazonaws.com
> ```

- Perluas ukuran volume boot.
- Setel kata sandi administrator.

Untuk mengonfigurasi pengaturan inisialisasi

1. Saat ingin mengonfigurasikan instans, buka file berikut ini dalam editor teks: `C:\ProgramData\Amazon\EC2-Windows\Launch\Config\LaunchConfig.json`.
2. Perbarui pengaturan berikut sesuai kebutuhan dan simpan perubahan Anda. Sediakan kata sandi dalam `adminPassword` hanya jika `adminPasswordtype` adalah `Specify`.

```
{
 "setComputerName": false,
 "setMonitorAlwaysOn": true,
 "setWallpaper": true,
 "addDnsSuffixList": true,
 "extendBootVolumeSize": true,
 "handleUserData": true,
 "adminPasswordType": "Random | Specify | DoNothing",
 "adminPassword": "password that adheres to your security policy (optional)"
}
```

Jenis kata sandi ditentukan sebagai berikut:

`Random`

EC2Launch menghasilkan dan mengenkripsikan kata sandi menggunakan kunci pengguna. Sistem akan menonaktifkan pengaturan ini setelah instans dijalankan sehingga kata sandi ini akan tetap ada jika instans tersebut di-boot ulang atau dihentikan dan dimulai.

`Specify`

EC2Launch menggunakan kata sandi yang Anda tentukan di `adminPassword`. Jika kata sandi tidak memenuhi persyaratan sistem, maka EC2Lauch membuat kata sandi acak sebagai gantinya. Kata sandi disimpan di `LaunchConfig.json` sebagai teks polos dan dihapus setelah Sysprep mengatur kata sandi administrator. EC2Launch mengenkripsi kata sandi menggunakan kunci pengguna.

`DoNothing`

EC2Launch menggunakan kata sandi yang Anda tentukan di file `unattend.xml`. Jika Anda tidak menentukan kata sandi di `unattend.xml`, akun administrator akan dinonaktifkan.

3. Di Windows PowerShell, jalankan perintah berikut untuk menjadwalkan skrip untuk dijalankan sebagai Tugas Terjadwal Windows. Skrip berjalan satu kali selama boot berikutnya dan kemudian menonaktifkan tugas-tugas ini agar tidak berjalan lagi.

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -Schedule
```

Jadwalkan EC2Launch untuk dijalankan di setiap boot

Anda bisa menjadwalkan EC2Launch untuk dijalankan pada setiap boot, bukan hanya pada boot awal.

Untuk memungkinkan EC2Launch dijalankan di setiap boot:

1. Buka Windows PowerShell dan jalankan perintah berikut:

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
SchedulePerBoot
```

2. Atau, jalankan executable dengan perintah berikut:

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Settings\Ec2LaunchSettings.exe
```

Kemudian pilih `Run EC2Launch on every boot`. Anda dapat menentukan apakah instans EC2 Anda `Shutdown without Sysprep` atau `Shutdown with Sysprep`.

> **Note**
>
> Saat Anda memungkinkan EC2Launch untuk dijalankan di setiap boot, perubahan berikut akan dilakukan setiap kali EC2Launch berjalan:
>
> - Jika `AdminPasswordType` masih diatur ke `Random`, EC2Launch akan menghasilkan kata sandi baru pada boot berikutnya. Setelah boot itu, `AdminPasswordType` secara otomatis diatur ke `DoNothing` untuk mencegah EC2launch menghasilkan kata sandi baru pada boot berikutnya. Untuk mencegah EC2launch membuat kata sandi baru pada boot pertama, atur `AdminPasswordType` ke `DoNothing` secara manual sebelum Anda reboot.
> - `HandleUserData` akan diatur kembali ke `false` kecuali data pengguna mengatur `persist` ke `true`. Untuk informasi selengkapnya, lihat the section called “Skrip data pengguna”.

Inisialisasi drive dan petakan huruf drive

Tentukan pengaturan di file `DriveLetterMappingConfig.json` untuk memetakan huruf drive ke volume pada instans EC2 Anda. Skrip menginisialisasi drive yang belum diinisialisasi dan dipartisi. Untuk informasi selengkapnya tentang mendapatkan detail volume di Windows, lihat Get-Volume di dokumentasi Microsoft.

Untuk memetakan huruf drive ke volume

1. Buka file `C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json` di editor teks.
2. Tentukan pengaturan volume berikut dan simpan perubahan Anda:

```
{
 "driveLetterMapping": [
  {
   "volumeName": "sample volume",
   "driveLetter": "H"
  }
 ]
```

```
}
```

3. Buka Windows PowerShell dan gunakan perintah berikut untuk menjalankan skrip EC2launch yang menginisialisasi disk:

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
```

Untuk menginisialisasi disk setiap kali booting instans, tambahkan bendera `-Schedule` sebagai berikut:

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -Schedule
```

Kirim log peristiwa Windows ke log konsol EC2.

Tentukan pengaturan di file `EventLogConfig.json` untuk mengirim log Peristiwa Windows ke log konsol EC2.

Untuk mengonfigurasi pengaturan untuk mengirim log Peristiwa Windows

1. Pada instans, buka file `C:\ProgramData\Amazon\EC2-Windows\Launch\Config\EventLogConfig.json` dalam editor teks.
2. Konfigurasikan pengaturan log berikut dan simpan perubahan Anda:

```
{
 "events": [
  {
   "logName": "System",
   "source": "An event source (optional)",
   "level": "Error | Warning | Information",
   "numEntries": 3
  }
 ]
}
```

3. Di Windows PowerShell, jalankan perintah berikut sehingga sistem menjadwalkan skrip untuk dijalankan sebagai Tugas Terjadwal Windows setiap kali instance di-boot.

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendEventLogs.ps1 -Schedule
```

Log dapat membutuhkan waktu tiga menit atau lebih untuk muncul di log konsol EC2.

### Kirim pesan Windows siap setelah boot berhasil

Layanan EC2Config mengirim pesan "Windows siap" ke konsol EC2 setelah setiap boot. EC2Launch mengirim pesan ini hanya setelah boot awal. Untuk kompatibilitas mundur dengan file layanan EC2Config, Anda dapat menjadwalkan EC2Launch untuk mengirim pesan ini setelah setiap boot. Pada contoh, buka Windows PowerShell dan jalankan perintah berikut. Sistem menjadwalkan skrip untuk dijalankan sebagai Tugas Terjadwal Windows.

```
C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\SendWindowsIsReady.ps1 -Schedule
```

### Riwayat versi EC2Launch

Untuk mengunduh dan menginstal EC2Launch versi terbaru, lihat Instal EC2Launch versi terbaru.

Tabel berikut menjelaskan versi EC2Launch yang dirilis.

| Versi | Detail | Tanggal rilis |
|---|---|---|
| 1.4.299 | • Logika yang dioptimalkan untuk mengambil versi EC2WinUtil driver.<br>• Mengurangi waktu untuk Windows Siap dengan menghapus kueri Hyper-V status. | 3 Maret 2026 |
| 1.4.183 | • Logika penginstal yang diperbarui untuk mencegah instalasi yang tidak didukung di Windows Server 2022.<br>• Diperbarui untuk mencetak versi EC2WinUtil driver ke log konsol instance. | 4 Februari 2026 |
| 1.4.6 | • Pesan log agen yang diperbarui. | 13 Oktober 2025 |
| 1.3.2005119 | • | 11 Februari 2025 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| | Memperbaiki masalah yang `Invoke-Userdata` akan gagal saat dipanggil tanpa parameter apa pun. | |
| 1.3.2005065 | • Memperbaiki masalah di mana informasi sertifikat RDP tidak diambil atau divalidasi dengan benar. Menambahkan fungsionalitas untuk secara otomatis memulai Layanan Desktop Jarak Jauh jika diperlukan. | 22 Oktober 2024 |
| 1.3.2005008 | • Diperbarui `Set-Wallpaper` untuk kembali ke latar belakang warna solid jika gambar wallpaper default tidak ditemukan. | 6 Agustus 2024 |
| 1.3.2004959 | • Logika penginstal yang diperbarui untuk mencegah instalasi yang tidak didukung pada Windows Server 2025 atau yang lebih baru. | 2 Juli 2024 |
| 1.3.2004891 | • Memperbaiki masalah yang tidak `HandleUserData` disetel `false` seperti yang diharapkan.<br>• Menambahkan opsi `Encrypted` kata sandi ke`LaunchConfig.json` .<br>• Mengubah `Settings UI` perilaku untuk mengenkripsi kata sandi yang ditentukan pengguna secara default.<br>• Ditambahkan `SetAdminPasswordConfig.ps1` untuk mengonversi opsi `Specify` kata `Encrypted` sandi ke opsi kata sandi di file konfigurasi agen. | 31 Mei 2024 |
| 1.3.2004617 | • Memperbaiki kesalahan saat mengatur wallpaper. | 15 Januari 2024 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.2004592 | • Izin akses yang diperbarui yang ditetapkan oleh install.ps1 untuk `%ProgramData%\Amazon\EC2-Windows\Launch` .<br>• Akses EC2Launch folder/file terbatas untuk baca-eksekusi hanya untuk akun pengguna standar.<br>• Mengubah agen agar berhenti menunggu Layanan Metadata Instans (IMDS) untuk diinisialisasi jika IMDS tidak diaktifkan untuk instans.<br>• Menambahkan batas waktu lima menit saat menunggu IMDS diinisialisasi.<br>• Mengubah agen untuk menulis telemetri ke log konsol instans sebelum pesan `Windows is Ready`, bukan setelahnya.<br>• Menambahkan dukungan wallpaper ke beberapa tipe instans baru.<br><br>Untuk informasi selengkapnya tentang izin akses dan izin akun pengguna direktori EC2launch, lihat. the section called “Struktur direktori EC2Launch” | 2 Januari 2024 |
| 1.3.20044491 | • Menambahkan telemetri untuk memantau penggunaan opsi Tentukan kata sandi admin. | 9 November 2023 |
| 1.3.20044462 | • Menambahkan flush setelah setiap penulisan ke konsol serial. | 18 Oktober 2023 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.2004438 | • Membatasi devolusi nama domain berdasarkan entri registri: `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel` .<br>• Izin `UserdataExecution.log` terbatas hanya untuk `Administrators` .<br>• Menambahkan pesan kesalahan di Log Peristiwa Windows saat inisialisasi log gagal. | 4 Oktober 2023 |
| 1.3.2004256 | • Menambahkan nilai `EnableSCSIPersistentReservations` ke log konsol.<br>• Menambahkan kemampuan coba lagi untuk Get-ConsolePort. | 7 Juli 2023 |
| 1.3.2004052 | • Memperbaiki kesalahan yang terjadi saat tidak ada kunci SSH yang ditentukan saat peluncuran instans.<br>• Diperbarui untuk mencoba lagi memulai layanan AmazonSSM Agent Windows saat mengalami kegagalan.<br>• Diperbarui menjadi gagal SysprepInstance.ps1 jika BeforeSysprep.cmd gagal dengan kode keluar bukan nol. | 8 Maret 2023 |
| 1.3.2003975 | • Memperbaiki masalah yang memengaruhi build Packer AMI di mana `SysprepInstance.ps1` mengembalikan 1. `$LastErrorCode` | 24 Desember 2022 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.2003961 | • Memperbaiki masalah di mana kata sandi administrator yang ditentukan secara eksplisit ditimpa dengan kata sandi acak pada instans yang diluncurkan dengan cepat.<br>• Memperbaiki masalah di mana Agen SSM gagal memulai pada tipe instans yang lebih kecil.<br>• Memperbaiki masalah saat log konsol instans berisi, `RDPCERTIFICATE-THUMBPRINT: 0000000 0000000000000000` bukan nilai cap jempol sertifikat RDP yang valid. | 6 Desember 2022 |
| 1.3.2003923 | • Memperbaiki logika untuk menemukan adaptor jaringan ketika PnPDeviceID kosong. | 9 November 2022 |
| 1.3.2003919 | • Diperbarui Get-ConsolePort untuk menggunakan informasi segmen PCI.<br>• Memperbaiki masalah di mana adaptor jaringan yang salah dapat dipilih setelah reboot.<br>• Logika batas SSM-Agent waktu mulai tetap.<br>• Memperbaiki kompatibilitas mundur untuk alias Send-AdminCredentials fungsi. | 8 November 2022 |
| 1.3.2003857 | • Memprioritaskan adaptor dengan gateway default saat adaptor jaringan utama dipilih.<br>• Enkripsi kata sandi dalam memori yang diperluas. | 3 Oktober 2022 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.2003824 | • Memperbaiki kesalahan selama `setComputerName` .<br>• Menambahkan logika untuk melewati aktivasi Windows ketika kode penagihan BYOL terdeteksi.<br>• Menambahkan enkripsi kata sandi dalam memori.<br>• Memperbaiki kesalahan selama inisialisasi volume aktif. `m6id.4xlarge` | 30 Agustus 2022 |
| 1.3.2003691 | • Logika tunggu IMDS yang diperbarui untuk membuat permintaan IMDSv2 saja.<br>• Memperbaiki bug yang memengaruhi instalasi eGPU. | 21 Juni 2022 |
| 1.3.2003639 | • Menambahkan logika tunggu adaptor jaringan untuk mencegah penggunaan sebelum inisialisasi.<br>• Memperbaiki masalah kecil. | 10 Mei 2022 |
| 1.3.2003498 | • Menambahkan telemetri.<br>• Menambahkan pintasan ke UI Pengaturan.<br>• PowerShell Skrip yang diformat.<br>• Memperbaiki masalah dengan shutdown yang terjadi sebelum BeforeSysprep.cmd selesai. | 31 Januari 2022 |
| 1.3.2003411 | Logika pembuatan kata sandi untuk mengecualikan kata sandi dengan kompleksitas rendah diubah. | 4 Agustus 2021 |
| 1.3.2003364 | Diperbarui Install-EgpuManager dengan dukungan IMDSv2. | 7 Juni 2021 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.2003312 | • Menambahkan baris log sebelum dan sesudah pengaturan `setMonitorAlwaysOn` .<br>• Menambahkan versi paket AWS Nitro Enclaves ke log konsol. | 4 Mei 2021 |
| 1.3.2003284 | Peningkatan model izin dengan memperbarui lokasi untuk menyimpan data pengguna ke `LocalAppData` . | 23 Maret 2021 |
| 1.3.2003236 | • Metode yang diperbarui untuk mengatur kata sandi pengguna di `Set-AdminAccount` dan `Randomize-LocalAdminPassword` .<br>• Memperbaiki`InitializeDisks` untuk memeriksa apakah disk diatur untuk membaca hanya sebelum mengaturnya menjadi dapat ditulis. | 11 Februari 2021 |
| 1.3.2003210 | Perbaiki lokalisasi untuk `install.ps1` . | 7 Januari 2021 |
| 1.3.2003205 | Perbaikan keamanan untuk `install.ps1` untuk memperbarui izin di direktori `%ProgramData%AmazonEC2-WindowsLaunchModuleScripts` . | 28 Desember 2020 |
| 1.3.2003189 | Menambahkan `w32tm resync` setelah menambahkan rute. | 4 Desember 2020 |
| 1.3.2003155 | Informasi tipe instans yang diperbarui. | 25 Agustus 2020 |
| 1.3.2003150 | Menambahkan `OsCurrentBuild` dan `OsReleaseId` ke output konsol . | 22 April 2020 |
| 1.3.2003040 | Memperbaiki logika fallback IMDS versi 1. | 7 April 2020 |
| 1.3.2002730 | Menambahkan dukungan untuk IMDS V2. | 3 Maret 2020 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.2002240 | Memperbaiki masalah kecil. | 31 Oktober 2019 |
| 1.3.2001660 | Memperbaiki masalah login otomatis untuk pengguna tanpa kata sandi setelah pertama kali menjalankan Sysprep. | 2 Juli 2019 |
| 1.3.2001360 | Memperbaiki masalah kecil. | 27 Maret 2019 |
| 1.3.2001220 | Semua PowerShell skrip ditandatangani. | 28 Februari 2019 |
| 1.3.2001200 | Memperbaiki masalah saat InitializeDisks.ps1 menjalankan skrip pada node di Cluster Failover Windows Server akan memformat drive pada node jarak jauh yang huruf drive-nya cocok dengan huruf drive lokal. | 27 Februari 2019 |
| 1.3.2001160 | Memperbaiki wallpaper yang hilang di Windows 2019. | 22 Februari 2019 |
| 1.3.2001040 | • Menambahkan plugin untuk mengatur monitor agar tidak pernah mati untuk memperbaiki masalah ACPI.<br>• Edisi dan versi SQL Server ditulis ke konsol. | 21 Januari 2019 |
| 1.3.2000930 | Perbaikan untuk menambahkan rute ke metadata pada ENI ipv6-enabled. | 2 Januari 2019 |
| 1.3.2000760 | • Menambahkan konfigurasi default untuk pengaturan RSS dan Menerima Antrean untuk perangkat ENA.<br>• Hibernasi dinonaktifkan selama Sysprep. | 5 Desember 2018 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.2000630 | • Ditambahkan rute 169.254.169. 253/32 untuk server DNS.<br>• Menambahkan filter pengaturan pengguna Admin.<br>• Perbaikan dilakukan pada hibernasi instans.<br>• Menambahkan opsi untuk menjadwalkan EC2Launch yang akan dijalankan di setiap boot. | 9 November 2018 |
| 1.3.200430.0 | • Ditambahkan rute 169.254.169. 123/32 ke layanan waktu AMZN.<br>• Ditambahkan rute 169.254.169. 249/32 ke layanan lisensi GRID.<br>• Menambahkan batas waktu habis 25 detik saat mencoba memulai Systems Manager. | 19 September 2018 |
| 1.3.200039.0 | • Memperbaiki huruf drive yang tidak tepat untuk volume EBS NVME.<br>• Menambahkan logging tambahan untuk versi driver NVME. | 15 Agustus 2018 |
| 1.3.2000080 | Memperbaiki masalah kecil. | |
| 1.3.610 | Memperbaiki masalah pengalihan output dan error ke file dari data pengguna. | |
| 1.3.590 | • Menambahkan tipe instans yang hilang di wallpaper.<br>• Memperbaiki masalah pemetaan huruf drive dan penginstalan disk. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.3.580 | • Memperbaiki `Get-Metadata` untuk menggunakan pengaturan proxy sistem default untuk permintaan web.<br>• Kasus khusus untuk NVMe dalam inisialisasi disk ditambahkan.<br>• Memperbaiki masalah kecil. | |
| 1.3.550 | Menambahkan opsi `-NoShutdown` untuk mengaktifkan Sysprep tanpa pematian. | |
| 1.3.540 | Memperbaiki masalah kecil. | |
| 1.3.530 | Memperbaiki masalah kecil. | |
| 1.3.521 | Memperbaiki masalah kecil. | |
| 1.3.0 | • Memperbaiki masalah panjang heksadesimal untuk perubahan nama komputer.<br>• Memperbaiki kemungkinan loop reboot untuk perubahan nama komputer.<br>• Memperbaiki masalah dalam pengaturan wallpaper. | |
| 1.2.0 | • Perbarui untuk menampilkan informasi tentang sistem operasi (OS) yang diinstal di log sistem EC2.<br>• Perbarui untuk menampilkan versi EC2Launch dan SSM Agent di log sistem EC2.<br>• Memperbaiki masalah kecil. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 1.1.2 | • Perbarui untuk menampilkan informasi driver ENA di log sistem EC2.<br>• Perbarui untuk mengecualikan Hyper-V dari logika filter NIC primer.<br>• Menambahkan AWS KMS server dan port ke kunci registri untuk aktivasi KMS.<br>• Penyiapan wallpaper yang ditingkatkan untuk banyak pengguna.<br>• Perbarui untuk menghapus rute dari penyimpanan persisten.<br>• Perbarui untuk menghapus z dari Availability Zone dalam daftar akhiran DNS.<br>• Perbarui untuk mengatasi masalah dengan tag <run AsLocalSystem > dalam data pengguna. | |
| 1.1.1 | Pelepasan awal. | |

## Gunakan layanan EC2config untuk melakukan tugas selama peluncuran instance sistem operasi Windows lama EC2

> **Note**
>
> EC2config telah mencapai akhir dukungan. Versi sistem operasi yang dijalankannya tidak lagi didukung oleh Microsoft. Kami sangat menyarankan Anda meningkatkan ke agen peluncuran terbaru.
> Agen peluncuran terbaru untuk Windows Server 2022 dan versi sistem operasi yang lebih baru adalah EC2launch v2, yang menggantikan EC2config dan EC2launch, dan sudah diinstal sebelumnya di AWS AMI Windows Server 2022 dan 2025. Anda juga dapat

menginstal dan mengkonfigurasi agen secara manual di Windows Server 2016 dan 2019. Untuk informasi selengkapnya, lihat Instal EC2launch v2.

AMI Windows untuk versi Windows Server sebelum Windows Server 2016 menyertakan layanan opsional, layanan EC2config ()`EC2Config.exe`. EC2Config dimulai saat instans melakukan boot dan menjalankan tugas selama pemulaian dan setiap kali Anda menghentikan atau memulai instans. EC2Config juga dapat melakukan tugas sesuai permintaan. Beberapa dari tugas ini diaktifkan secara otomatis, sementara yang lainnya harus diaktifkan secara manual. Meskipun opsional, layanan ini menyediakan akses ke fitur lanjutan yang tidak tersedia tanpa layanan ini. Layanan ini berjalan diLocalSystem Akun.

Layanan EC2Config menjalankan Sysprep, alat Microsoft yang memungkinkan Anda untuk membuat AMI Windows kustom yang dapat digunakan kembali. Saat EC2Config memanggil Sysprep, file digunakan di `%ProgramFiles%\Amazon\EC2ConfigService\Settings` untuk menentukan operasi mana yang akan dilakukan. Anda dapat mengedit file-file ini secara tidak langsung menggunakan dialog sistem Properti Layanan EC2, atau langsung menggunakan editor XML atau editor teks. Namun, ada beberapa pengaturan lanjutan yang tidak tersedia di dialog sistem Properti Layanan Ec2, jadi Anda harus mengedit entri tersebut secara langsung.

Jika Anda membuat AMI dari sebuah instans setelah memperbarui pengaturannya, pengaturan baru tersebut diterapkan ke setiap instans yang diluncurkan dari AMI baru. Untuk informasi tentang membuat grafik, lihat Buat EBS-backed AMI Amazon.

EC2Config menggunakan file pengaturan untuk mengontrol operasinya. Anda dapat memperbarui file pengaturan ini dengan menggunakan alat grafis atau dengan mengedit file XML secara langsung. Biner layanan dan file tambahan terdapat dalam direktori `%ProgramFiles%\Amazon\EC2ConfigService`.

Daftar Isi


- EC2config dan AWS Systems Manager
- Tugas EC2Config
- File pengaturan EC2Config
- Menginstal EC2Config versi terbaru
- Mengonfigurasi setelan proksi.NET untuk layanan EC2Config
- Setel properti layanan EC2config dari dialog sistem pada instance Windows EC2 Anda
- Memecahkan masalah dengan agen peluncuran EC2config

- Riwayat versi EC2Config

### EC2config dan AWS Systems Manager

Layanan EC2Config memproses permintaan Systems Manager pada instans yang dibuat dari AMI untuk versi Windows Server sebelum Windows Server 2016 yang diterbitkan sebelum November 2016.

Instans dibuat dari AMI untuk versi Windows Server sebelum Windows Server 2016 yang diterbitkan setelah November 2016 menyertakan layanan EC2Config dan SSM Agent. EC2Config melakukan semua tugas yang dijelaskan sebelumnya, dan SSM Agent memproses permintaan untuk kapabilitas Systems Manager seperti Run Command dan State Manager.

Anda dapat menggunakan Run Command untuk memutakhirkan instans yang ada untuk digunakan pada layanan EC2Config dan SSM Agent versi terbaru. Untuk informasi selengkapnya, lihat Memperbarui Agen SSM menggunakan Run Command di Panduan AWS Systems Manager Pengguna.

### Tugas EC2Config

EC2Config menjalankan tugas startup saat instans dimulai pertama kali dan kemudian menonaktifkannya. Untuk menjalankan tugas ini lagi, Anda harus secara eksplisit mengaktifkannya sebelum mematikan instans, atau dengan menjalankan Sysprep secara manual. Tugas-tugas tersebut adalah sebagai berikut:

- Tetapkan kata sandi terenkripsi acak untuk akun administrator.
- Buat dan instal sertifikat host yang digunakan untuk Remote Desktop Connection.
- Secara dinamis, perluas partisi sistem operasi untuk menyertakan ruang yang tidak dipartisi.
- Jalankan data pengguna yang ditentukan (dan Cloud-Init, jika diinstal). Untuk informasi selengkapnya tentang menentukan data pengguna, lihat Jalankan perintah saat Anda meluncurkan instance EC2 dengan input data pengguna.

EC2Config melakukan tugas-tugas berikut setiap kali instans dimulai:

- Ubah nama host agar sesuai dengan alamat IP pribadi dalam notasi Hex (tugas ini dinonaktifkan secara default dan harus diaktifkan untuk dijalankan pada awal instance).
- Konfigurasikan server manajemen kunci (AWS KMS), periksa status aktivasi Windows, dan aktifkan Windows seperlunya.

- Pasang semua volume Amazon EBS dan volume penyimpanan instans, dan petakan nama volume ke huruf drive.
- Tulis entri log peristiwa ke konsol untuk membantu pemecahan masalah (tugas ini dinonaktifkan secara default dan harus diaktifkan untuk dijalankan pada awal instans).
- Tulis ke konsol bahwa Windows sudah siap.
- Tambahkan rute khusus ke adaptor jaringan utama untuk mengaktifkan alamat IP berikut jika ada satu NIC atau banyak NIC dilampirkan: `169.254.169.250`, `169.254.169.251`, dan`169.254.169.254`. Alamat ini digunakan oleh Windows Activation dan ketika Anda mengakses metadata instans.

> **Note**
>
> Jika OS Windows dikonfigurasi untuk menggunakan IPv4, alamat link-lokal IPv4 ini dapat digunakan. Jika OS Windows memiliki tumpukan protokol jaringan IPv4 dinonaktifkan dan menggunakan IPv6 sebagai gantinya, tambahkan `[fd00:ec2::250]` sebagai pengganti dan. `169.254.169.250` `169.254.169.251` Kemudian, tambahkan `[fd00:ec2::254]` sebagai pengganti `169.254.169.254`.

EC2Config melakukan tugas berikut setiap kali pengguna masuk:

- Tampilkan informasi wallpaper ke latar belakang desktop.

Saat instans sedang berjalan, Anda dapat meminta agar EC2Config melakukan tugas berikut sesuai permintaan:

- Jalankan Sysprep dan matikan instans sehingga Anda dapat membuat AMI darinya. Untuk informasi selengkapnya, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.

File pengaturan EC2Config

File pengaturan mengontrol operasi file layanan EC2Config. File-file ini terletak di direktori `C:\Program Files\Amazon\Ec2ConfigService\Settings`:

- `ActivationSettings.xml`—Mengontrol aktivasi produk menggunakan server manajemen kunci (AWS KMS).

- `AWS.EC2.Windows.CloudWatch.json`—Mengontrol penghitung kinerja mana yang akan dikirim ke CloudWatch dan log mana yang akan dikirim ke CloudWatch Log.
- `BundleConfig.xml`—Mengontrol bagaimana EC2Config mempersiapkan instans yang didukung penyimpanan instans untuk pembuatan AMI.
- `Config.xml`—Mengontrol pengaturan utama.
- `DriveLetterConfig.xml`—Mengontrol pemetaan huruf drive.
- `EventLogConfig.xml`—Mengontrol informasi log peristiwa yang ditampilkan di konsol saat instans sedang booting.
- `WallpaperSettings.xml`—Mengontrol informasi yang ditampilkan di latar belakang desktop.

### ActivationSettings.xml

File ini berisi pengaturan yang mengontrol aktivasi produk. Saat Windows melakukan boot, file layanan EC2Config memeriksa apakah Windows sudah diaktifkan. Jika Windows belum diaktifkan, maka Windows mencoba mengaktifkan Windows dengan mencari server AWS KMS yang ditentukan.

- `SetAutodiscover`—Menunjukkan apakah akan mendeteksi AWS KMS secara otomatis.
- `TargetKMSServer`—Menyimpan alamat IP pribadi dari a AWS KMS. AWS KMS harus berada di Wilayah yang sama dengan instans Anda.
- `DiscoverFromZone`—Menemukan AWS KMS server dari zona DNS yang ditentukan.
- `ReadFromUserData`—Mendapat AWS KMS server dari UserData.
- `LegacySearchZones`—Menemukan AWS KMS server dari zona DNS yang ditentukan.
- `DoActivate`—Mencoba aktivasi menggunakan pengaturan tertentu di bagian. Nilai ini bisa jadi `true` atau `false`.
- `LogResultToConsole`—Menampilkan hasil ke konsol.

### BundleConfig.xml

File ini berisi pengaturan yang mengontrol bagaimana EC2Config mempersiapkan sebuah instans untuk pembuatan AMI.

- `AutoSysprep`—Menunjukkan apakah akan menggunakan Sysprep secara otomatis. Ubah nilainya menjadi `Yes` untuk menggunakan Sysprep.

- `SetRDPCertificate`—Mengatur sertifikat yang ditandatangani sendiri ke server Remote Desktop. Ini memungkinkan Anda untuk melakukan RDP dengan aman ke dalam instans. Ubah nilainya menjadi `Yes` jika instans baru harus memiliki sertifikat.

  Pengaturan ini tidak digunakan untuk instans dengan versi sistem operasi sebelum Windows Server 2016, karena mereka dapat menghasilkan sertifikat mereka sendiri.
- `SetPasswordAfterSysprep`—Mengatur kata sandi acak pada instans yang baru diluncurkan, mengenkripsinya dengan kunci peluncuran pengguna, dan menghasilkan kata sandi terenkripsi ke konsol. Ubah nilai pengaturan ini ke `No` jika instans baru tidak boleh diatur ke kata sandi terenkripsi acak.

Config.xml

Plug-ins

- `Ec2SetPassword`—Membuat sandi terenkripsi acak setiap kali Anda meluncurkan sebuah instans. Fitur ini dinonaktifkan secara default setelah peluncuran pertama sehingga reboot instans ini tidak mengubah sandi yang ditetapkan oleh pengguna. Ubah pengaturan ini menjadi `Enabled` untuk terus menghasilkan sandi setiap kali Anda meluncurkan sebuah instans.

  Pengaturan ini penting jika Anda berencana membuat AMI dari instans Anda.
- `Ec2SetComputerName`—Mengatur nama host instans menjadi nama unik berdasarkan alamat IP instans dan melakukan boot ulang instans. Untuk mengatur nama host Anda sendiri, atau mencegah perubahan nama host yang ada, jangan aktifkan pengaturan ini.
- `Ec2InitializeDrives`—Menginisialisasi dan memformat semua volume selama startup. Fitur ini diaktifkan secara default.
- `Ec2EventLog`—Menampilkan entri log peristiwa di konsol. Secara default, tiga entri kesalahan terbaru dari log aktivitas sistem akan ditampilkan. Untuk menentukan entri log peristiwa yang akan ditampilkan, edit file `EventLogConfig.xml` yang terletak di direktori `EC2ConfigService\Settings`. Untuk informasi tentang pengaturan dalam file ini, lihat Kunci Eventlog.
- `Ec2ConfigureRDP`—Menyiapkan sertifikat yang ditandatangani sendiri di instans, sehingga pengguna dapat mengakses instans dengan aman menggunakan Remote Desktop. Pengaturan ini tidak digunakan untuk instans dengan versi sistem operasi sebelum Windows Server 2016, karena mereka dapat menghasilkan sertifikat mereka sendiri.
- `Ec2OutputRDPCert`—Menampilkan informasi sertifikat Remote Desktop ke konsol, sehingga pengguna dapat memverifikasinya dengan sidik jari.

- `Ec2SetDriveLetter`—Mengatur huruf drive dari volume yang terpasang berdasarkan pengaturan yang ditentukan pengguna. Secara default, ketika dilampirkan ke sebuah instans, volume Amazon EBS dapat dipasang menggunakan huruf drive pada instans tersebut. Untuk menentukan pemetaan huruf drive Anda, edit file `DriveLetterConfig.xml` yang terletak di direktori `EC2ConfigService\Settings`.
- `Ec2WindowsActivate`—Plug-in menangani aktivasi Windows. Ia memeriksa untuk melihat apakah Windows diaktifkan. Jika tidak, itu memperbarui pengaturan AWS KMS klien, dan kemudian mengaktifkan Windows.

  Untuk mengubah AWS KMS pengaturan, edit `ActivationSettings.xml` file yang terletak di `EC2ConfigService\Settings` direktori.
- `Ec2DynamicBootVolumeSize`—Memperluas Disk 0/Volume 0 untuk menyertakan ruang yang tidak dipartisi.
- `Ec2HandleUserData`—Membuat dan menjalankan skrip yang dibuat oleh pengguna pada peluncuran pertama instans setelah Sysprep dijalankan. Perintah yang dibungkus tag skrip disimpan ke file batch, dan perintah yang dibungkus PowerShell tag disimpan ke file.ps1 (sesuai dengan kotak centang Data Pengguna pada dialog sistem Properti Layanan Ec2).
- `Ec2ElasticGpuSetup`—Memasang paket perangkat lunak Elastic GPU jika instans dikaitkan dengan GPU elastis.
- `Ec2FeatureLogging`—Mengirimkan instalasi fitur Windows dan status layanan yang sesuai ke konsol. Didukung hanya untuk Hyper-V fitur Microsoft dan layanan vmms yang sesuai.

Pengaturan Global

- `ManageShutdown`—Memastikan bahwa instans yang diluncurkan dari S3-backed AMI Amazon tidak berakhir saat menjalankan Sysprep.
- `SetDnsSuffixList`—Mengatur sufiks DNS adaptor jaringan untuk Amazon EC2. Hal ini memungkinkan resolusi DNS server yang berjalan di Amazon EC2 tanpa memberikan nama domain yang sepenuhnya memenuhi syarat.

> **Note**
>
> Ini menambahkan pencarian akhiran DNS untuk domain berikut dan mengonfigurasi akhiran standar lainnya. Untuk informasi selengkapnya tentang cara agen peluncuran mengatur sufiks DNS, lihat. Konfigurasikan Sufiks DNS untuk agen peluncuran EC2 Windows

```
region.ec2-utilities.amazonaws.com
```

- `WaitForMetaDataAvailable`—Memastikan bahwa layanan EC2Config akan menunggu metadata dapat diakses dan jaringan tersedia sebelum melanjutkan boot. Pemeriksaan ini memastikan bahwa EC2Config dapat memperoleh informasi dari metadata untuk aktivasi dan plug-in lainnya.
- `ShouldAddRoutes`—Menambahkan rute khusus ke adaptor jaringan utama untuk mengaktifkan alamat IP berikut jika ada beberapa NIC dilampirkan: 169.254.169.250, 169.254.169.251, dan 169.254.169.254. Alamat ini digunakan oleh Windows Activation dan ketika Anda mengakses metadata instans.
- `RemoveCredentialsfromSyspreponStartup`—Menghapus kata sandi administrator dari `Sysprep.xml` saat layanan dimulai lagi. Untuk memastikan bahwa kata sandi ini tetap ada, edit pengaturan ini.

### DriveLetterConfig.xml

File ini berisi pengaturan yang mengontrol pemetaan huruf drive. Secara default, volume dapat dipetakan ke huruf drive yang tersedia. Anda dapat memasang volume ke huruf drive tertentu sebagai berikut.

```
<?xml version="1.0" standalone="yes"?>
<DriveLetterMapping>
  <Mapping>
    <VolumeName></VolumeName>
    <DriveLetter></DriveLetter>
  </Mapping>
  . . .
  <Mapping>
    <VolumeName></VolumeName>
    <DriveLetter></DriveLetter>
  </Mapping>
</DriveLetterMapping>
```

- `VolumeName`—Label volume. Sebagai conto,*`My Volume`* Untuk menentukan pemetaan untuk volume penyimpanan instans, gunakan label `Temporary Storage X`, di mana `X` adalah angka dari 0 sampai 25.
- `DriveLetter`—Huruf drive. Sebagai conto,*`M:`* Pemetaan gagal jika huruf drive sudah digunakan.

EventLogConfig.xml

File ini berisi pengaturan yang mengontrol informasi log peristiwa yang ditampilkan di konsol saat instans sedang di-boot. Secara default, kami menampilkan tiga entri kesalahan terbaru dari log peristiwa Sistem.

- `Category`—Kunci log peristiwa yang akan dipantau.
- `ErrorType`—Tipe peristiwa (misalnya, `Error`, `Warning`, `Information`)
- `NumEntries`—Jumlah peristiwa yang disimpan untuk kategori ini.
- `LastMessageTime`—Untuk mencegah pesan yang sama didorong berulang kali, layanan memperbarui nilai ini setiap kali mendorong suatu pesan.
- `AppName`—Sumber peristiwa atau aplikasi yang mencatat peristiwa tersebut.

WallpaperSettings.xml

File ini berisi pengaturan yang mengontrol informasi yang ditampilkan di latar belakang desktop. Informasi berikut ini ditampilkan secara default.

- `Hostname`—Menampilkan nama komputer.
- `Instance ID`—Menampilkan ID instans.
- `Public IP Address`—Menampilkan alamat IP publik instans.
- `Private IP Address`—Menampilkan alamat IP privat instans.
- `Availability Zone`—Menampilkan Zona Ketersediaan tempat instans berjalan.
- `Instance Size`—Menampilkan tipe instans.
- `Architecture`—Menampilkan pengaturan variabel lingkungan `PROCESSOR_ARCHITECTURE`.

Anda dapat menghapus informasi apa pun yang ditampilkan secara default dengan menghapus entrinya. Anda dapat menambahkan metadata instans tambahan untuk ditampilkan sebagai berikut.

```
<WallpaperInformation>
  <name>display_name</name>
  <source>metadata</source>
  <identifier>meta-data/path</identifier>
</WallpaperInformation>
```

Anda dapat menambahkan variabel lingkungan Sistem tambahan untuk ditampilkan sebagai berikut.

```
<WallpaperInformation>
  <name>display_name</name>
  <source>EnvironmentVariable</source>
  <identifier>variable-name</identifier>
</WallpaperInformation>
```

InitializeDrivesSettings.xml

File ini berisi pengaturan yang mengontrol bagaimana EC2Config menginisialisasi drive.

Secara default, EC2Config menginisialisasi drive yang tidak dibawa online dengan sistem operasi. Anda dapat menyesuaikan plugin sebagai berikut.

```
<InitializeDrivesSettings>
    <SettingsGroup>setting</SettingsGroup>
</InitializeDrivesSettings>
```

Gunakan grup pengaturan untuk menentukan cara Anda ingin memulai drive:

FormatWithMEMOTONG

Mengaktifkan perintah TRIM saat memformat drive. Setelah drive diformat dan diinisialisasi, sistem memulihkan konfigurasi TRIM.

Dimulai dengan EC2Config versi 3.18, perintah TRIM dinonaktifkan selama operasi format disk secara default. Ini meningkatkan waktu pemformatan. Gunakan pengaturan ini untuk mengaktifkan TRIM selama operasi format disk untuk EC2Config versi 3.18 dan setelahnya.

FormatWithoutMEMOTONG

Menonaktifkan perintah TRIM saat memformat drive dan meningkatkan waktu pemformatan di Windows. Setelah drive diformat dan diinisialisasi, sistem mengembalikan konfigurasi TRIM.

DisableInitializeDrives

Menonaktifkan pemformatan untuk drive baru. Gunakan pengaturan ini untuk memulai drive secara manual.

## Menginstal EC2Config versi terbaru

> **Note**
>
> Agen peluncuran terbaru untuk Windows Server 2022 dan versi sistem operasi yang lebih baru adalah EC2launch v2, yang menggantikan EC2config dan EC2launch. EC2launch v2 hadir pra-instal di AMI AWS Windows Server 2022 dan 2025. Anda juga dapat menginstal dan mengkonfigurasi agen secara manual di Windows Server 2016 dan 2019. Untuk informasi selengkapnya, lihat Instal EC2launch v2.

Untuk informasi tentang cara menerima notifikasi pembaruan EC2Config, lihat Berlangganan pemberitahuan agen peluncuran EC2 Windows. Untuk informasi tentang perubahan di setiap versi, lihat Riwayat versi EC2Config.

### Sebelum Anda memulai

- Pastikan Anda memiliki.NET framework 3.5 SP1 atau yang lebih tinggi
- Secara default, Setup mengganti file pengaturan Anda dengan file pengaturan default selama instalasi dan memulai ulang layanan EC2Config saat instalasi selesai. Jika Anda mengubah pengaturan layanan EC2Config, salin file `config.xml` dari direktori `%Program Files%\Amazon\Ec2ConfigService\Settings`. Setelah Anda memperbarui layanan EC2Config, Anda dapat memulihkan file ini untuk mempertahankan perubahan konfigurasi Anda.

### Verifikasi versi EC2Config

Gunakan prosedur berikut untuk memverifikasi versi EC2Config yang diinstal pada instans Anda.

Untuk memverifikasi versi EC2Config yang diinstal

1. Luncurkan sebuah instans dari AMI dan hubungkan diri Anda dengan instans tersebut.
2. Di Panel Kontrol, pilih Program dan Fitur.
3. Dalam daftar program yang diinstal, cari `Ec2ConfigService`. Nomor versinya muncul di kolom Versi.

### Perbarui EC2Config

Gunakan prosedur berikut untuk mengunduh dan menginstal EC2Config versi terbaru pada instans Anda.

Untuk mengunduh dan menginstal EC2Config versi terbaru

1. Unduh dan unzip penginstal EC2Config.
2. Jalankan `EC2Install.exe`. Untuk daftar lengkap opsi, jalankan `EC2Install` dengan opsi `/?`. Secara default, penyiapan menampilkan perintah. Untuk menjalankan perintah tanpa prompt, gunakan`/quiet` pilihan.

   > **Important**
   >
   > Untuk mempertahankan pengaturan kustom dari file `config.xml` yang Anda simpan, jalankan `EC2Install` dengan opsi `/norestart`, pulihkan pengaturan Anda, dan kemudian mulai ulang layanan EC2Config secara manual.

3. Jika Anda sedang menjalankan EC2Config versi 4.0 atau setelahnya, maka Anda harus memulai ulang SSM Agent pada instans dari snap-in Layanan Microsoft.

   > **Note**
   >
   > Informasi versi EC2Config yang diperbarui tidak akan muncul di Log Sistem instans atau pemeriksaan Trusted Advisor hingga Anda melakukan boot ulang atau menghentikan dan memulai instans Anda.

Untuk mengunduh dan menginstal versi terbaru dari EC2config menggunakan PowerShell

Untuk mengunduh, unzip, dan menginstal versi terbaru EC2config menggunakan PowerShell, jalankan perintah berikut dari jendela: PowerShell

```
$Url = "https://s3.amazonaws.com/ec2-downloads-windows/EC2Config/EC2Install.zip"
$DownloadZipFile = "$env:USERPROFILE\Desktop\" + $(Split-Path -Path $Url -Leaf)
$ExtractPath = "$env:USERPROFILE\Desktop\"
Invoke-WebRequest -Uri $Url -OutFile $DownloadZipFile
$ExtractShell = New-Object -ComObject Shell.Application
$ExtractFiles = $ExtractShell.Namespace($DownloadZipFile).Items()
$ExtractShell.NameSpace($ExtractPath).CopyHere($ExtractFiles)
```

```
Start-Process $ExtractPath
Start-Process `
    -FilePath $env:USERPROFILE\Desktop\EC2Install.exe `
    -ArgumentList "/S"
```

> **Note**
>
> Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:
>
> ```
> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
> ```

Verifikasi instalasi dengan memeriksa `C:\Program Files\Amazon\` untuk direktori `Ec2ConfigService`.

Mengonfigurasi setelan proksi.NET untuk layanan EC2Config

Anda dapat mengonfigurasi layanan EC2Config untuk berkomunikasi melalui proxy menggunakan salah satu metode berikut: AWS SDK untuk.NET, `system.net` elemen, atau Kebijakan Grup Microsoft dan Internet Explorer. Menggunakan AWS SDK untuk .NET adalah metode yang disukai karena Anda dapat menentukan kredentif masuk.

Metode

- Konfigurasikan pengaturan proxy menggunakan AWS SDK untuk .NET (Lebih disukai)
- Konfigurasikan pengaturan proxy menggunakan elemen system.net
- Konfigurasikan pengaturan proxy menggunakan Kebijakan Grup Microsoft dan Internet Explorer Microsoft

Konfigurasikan pengaturan proxy menggunakan AWS SDK untuk .NET (Lebih disukai)

Anda dapat mengonfigurasi pengaturan proxy untuk layanan EC2Config dengan menentukan elemen `proxy` di file `Ec2Config.exe.config`. Untuk informasi selengkapnya, lihat Referensi File Konfigurasi untuk AWS SDK untuk.NET.

Untuk menentukan elemen proxy di Ec2Config.exe.config

1. Edit file `Ec2Config.exe.config` pada instans tempat Anda menginginkan file layanan EC2Config untuk berkomunikasi melalui proksi. Secara default, file terletak di direktori berikut: `%ProgramFiles%\Amazon\Ec2ConfigService`.
2. Tambahkan elemen `aws` berikut ini ke `configSections`. Jangan tambahkan ini ke `sectionGroups` yang ada.

   Untuk EC2Config versi 3.17 atau sebelumnya

   ```
   <configSections>
      <section name="aws" type="Amazon.AWSSection, AWSSDK"/>
   </configSections>
   ```

   Untuk EC2Config versi 3.18 atau setelahnya

   ```
   <configSections>
        <section name="aws" type="Amazon.AWSSection, AWSSDK.Core"/>
   </configSections>
   ```

3. Tambahkan elemen `aws` berikut ini ke file `Ec2Config.exe.config`.

   ```
   <aws>
      <proxy
        host="string value"
        port="string value"
        username="string value"
        password="string value" />
   </aws>
   ```

4. Simpan perubahan Anda.

Konfigurasikan pengaturan proxy menggunakan elemen system.net

Anda dapat menentukan pengaturan proxy di elemen `system.net` di file `Ec2Config.exe.config`. Untuk informasi selengkapnya, lihat elemen defaultProxy (pengaturan jaringan).

Untuk menentukan elemen system.net di Ec2Config.exe.config

1. Edit file `Ec2Config.exe.config` pada instans tempat Anda menginginkan file layanan EC2Config untuk berkomunikasi melalui proksi. Secara default, file terletak di direktori berikut: `%ProgramFiles%\Amazon\Ec2ConfigService`.
2. Tambahkan entri `defaultProxy` ke `system.net`. Untuk informasi selengkapnya, lihat elemen defaultProxy (pengaturan jaringan).

   Misalnya, konfigurasi berikut merutekan semua lalu lintas untuk menggunakan proxy yang saat ini dikonfigurasi untuk Internet Explorer, dengan pengecualian lalu lintas metadata dan lisensi, yang akan melewati proxy.

   ```
   <defaultProxy>
       <proxy usesystemdefault="true" />
       <bypasslist>
           <add address="169.254.169.250" />
           <add address="169.254.169.251" />
           <add address="169.254.169.254" />
           <add address="[fd00:ec2::250]" />
           <add address="[fd00:ec2::254]" />
       </bypasslist>
   </defaultProxy>
   ```
3. Simpan perubahan Anda.

Konfigurasikan pengaturan proxy menggunakan Kebijakan Grup Microsoft dan Internet Explorer Microsoft

Layanan EC2Config berjalan di bawah akun pengguna Sistem Lokal. Anda dapat menentukan pengaturan proksi seluruh instans untuk akun ini di Internet Explorer setelah Anda mengubah pengaturan Kebijakan Grup pada instans.

Untuk mengonfigurasi pengaturan proxy menggunakan Kebijakan Grup dan Internet Explorer

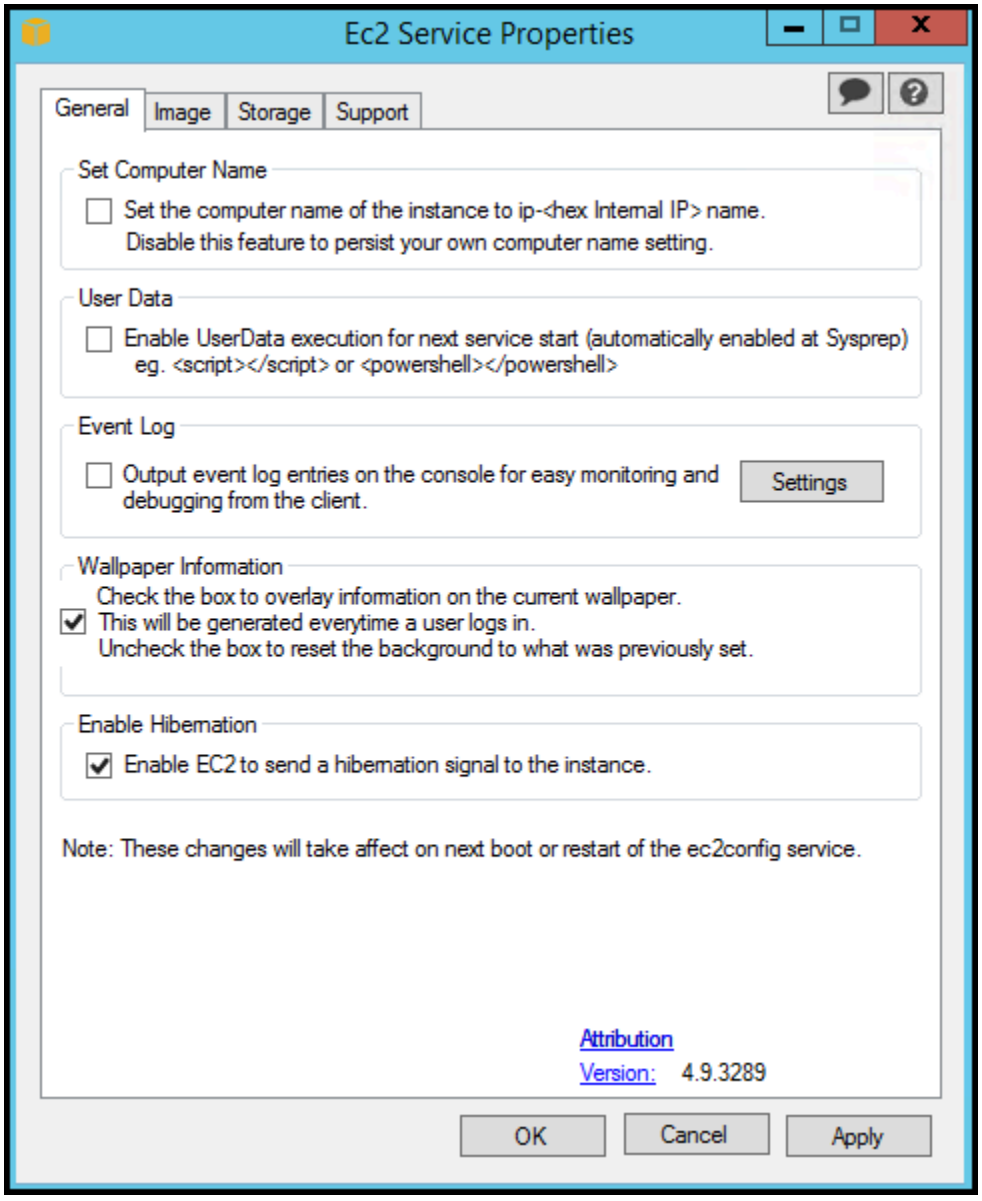
1. Pada instans di mana Anda menginginkan file layanan EC2Config berkomunikasi melalui proxy, buka Command prompt sebagai Administrator, ketik, **`gpedit.msc`**dan tekan Enter.
2. Di Editor Kebijakan Grup Lokal, di bawah Kebijakan Komputer Lokal, pilih Konfigurasi Komputer, Templat Administratif, Komponen Windows, Internet Explorer.

3. Di panel kanan, pilih Buat pengaturan proxy per mesin (bukan per pengguna) lalu pilih Edit pengaturan kebijakan.
4. Pilih Diaktifkan, lalu pilih Terapkan.
5. Buka Internet Explorer, lalu pilih tombol Alat.
6. Pilih Opsi Internet, lalu pilih tab Koneksi.
7. Pilih Pengaturan LAN.
8. Di bawah Server proxy, pilih opsi Gunakan server proxy untuk LAN Anda.
9. Tentukan alamat dan informasi port lalu pilih OK.

Setel properti layanan EC2config dari dialog sistem pada instance Windows EC2 Anda

Prosedur berikut menjelaskan cara menggunakan dialog sistem Pro perti Layanan EC2 untuk mengaktifkan atau menonaktifkan pengaturan.

1. Jalankan dan hubungkan ke instans Windows Anda.
2. Dari menu Mulai, pilih Semua Program, lalu pilih Peng EC2ConfigService aturan.




3. Pada tab Umum pada dialog sistem Properti Layanan EC2, Anda dapat mengaktifkan atau menonaktifkan pengaturan berikut.

   Atur Nama Komputer

   Jika pengaturan ini diaktifkan (dinonaktifkan secara default), nama host dibandingkan dengan alamat IP internal saat ini di setiap boot; jika nama host dan alamat IP internal tidak cocok, nama host disetel ulang untuk memuat alamat IP internal dan kemudian sistem melakukan boot ulang untuk mengambil nama host baru. Untuk mengatur nama host Anda sendiri, atau untuk mencegah perubahan nama host yang ada, jangan aktifkan pengaturan ini.

Data Pengguna

Eksekusi data pengguna memungkinkan Anda menentukan skrip dalam metadata instans. Secara default, skrip ini berjalan selama peluncuran awal. Anda juga dapat mengonfigurasinya untuk dijalankan saat Anda melakukan boot ulang atau memulai instans, atau setiap kali Anda melakukan boot ulang atau memulai instans.

Jika Anda memiliki skrip yang besar, kami menyarankan agar Anda menggunakan data pengguna untuk mengunduh skrip, dan kemudian menjalankannya.

Untuk informasi selengkapnya, lihat Eksekusi data pengguna.

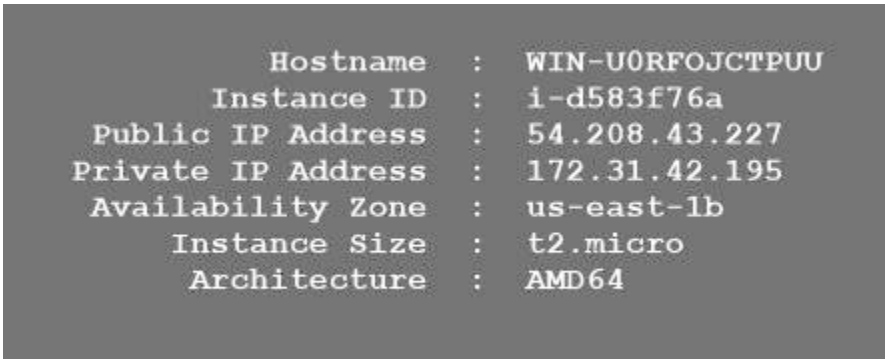
Log Peristiwa

Gunakan pengaturan ini untuk menampilkan entri log peristiwa di konsol selama boot untuk memudahkan pemantauan dan debugging.

Pilih Pengaturan untuk menentukan filter untuk entri log yang dikirim ke konsol. Filter default mengirimkan tiga entri kesalahan terbaru dari log peristiwa sistem ke konsol.

Informasi Wallpaper

Gunakan pengaturan ini untuk menampilkan informasi sistem di latar belakang desktop. Berikut ini adalah contoh informasi yang ditampilkan di latar belakang desktop.

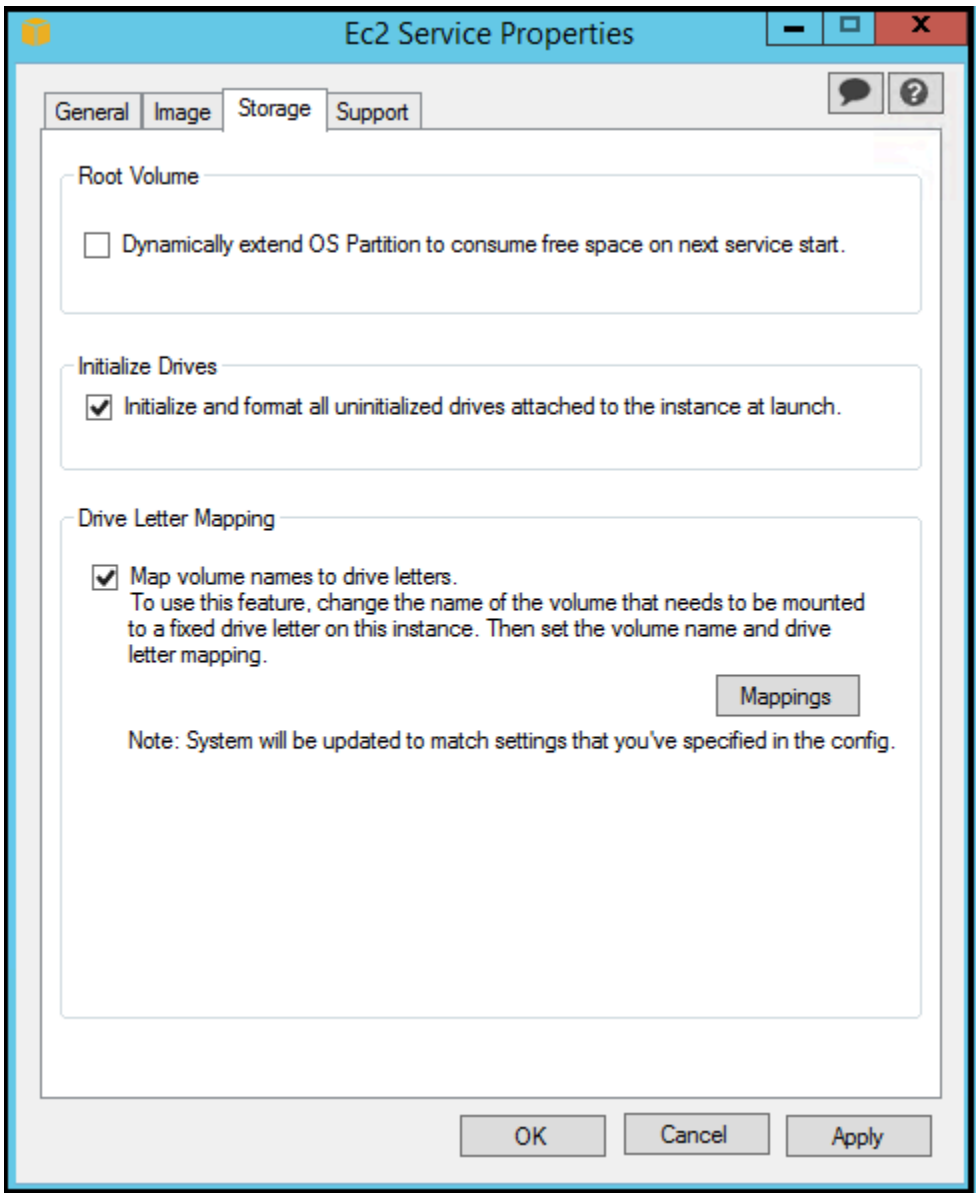



Informasi yang ditampilkan di latar belakang desktop dikontrol oleh file pengaturan `EC2ConfigService\Settings\WallpaperSettings.xml`.

Aktifkan Hibernasi

Gunakan pengaturan ini untuk memungkinkan EC2 memberi sinyal pada sistem operasi untuk melakukan hibernasi.

4. Pilih tab Penyimpanan. Anda dapat mengaktifkan atau menonaktifkan pengaturan berikut.




Volume root

Pengaturan ini secara dinamis memperluas Disk 0/Volume 0 untuk menyertakan ruang yang tidak dipartisi. Ini dapat berguna ketika instance di-boot dari volume root yang memiliki ukuran khusus.

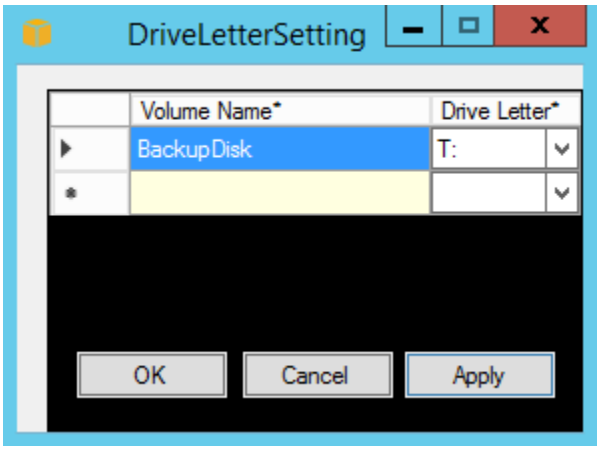
Inisialisasi Drive

Pengaturan ini memformat dan memasang semua volume yang terpasang ke instans selama memulai.

Pemetaan Huruf Drive

Sistem memetakan volume yang dilampirkan ke sebuah instans ke huruf drive. Untuk volume Amazon EBS, default-nya adalah menetapkan huruf drive dari D: ke Z:. Misalnya volume toko, defaultnya tergantung pada driver. AWS Driver PV dan driver Citrix PV menetapkan volume penyimpanan instance huruf drive dari Z: ke A:. Driver Red Hat menetapkan volume penyimpanan instans huruf drive dari D: ke Z:.

Untuk memilih huruf drive untuk volume Anda, pilih Pem etaan. Di kotak DriveLetterSetting dialog, tentukan nilai Nama Volume dan Drive Letter untuk setiap volume, pilih Terapkan, lalu pilih OK. Kami menganjurkan agar Anda memilih huruf drive yang menghindari konflik dengan huruf drive yang mungkin digunakan, seperti huruf drive di tengah alfabet.




Setelah Anda menentukan pemetaan huruf drive dan melampirkan volume dengan label yang sama dengan salah satu nama volume yang Anda tentukan, EC2Config secara otomatis menetapkan huruf drive yang Anda tentukan ke volume itu. Namun, pemetaan huruf kandar gagal jika huruf kandar sudah digunakan. Perhatikan bahwa EC2Config tidak mengubah huruf drive volume yang sudah dipasang saat Anda menentukan pemetaan huruf drive.

5. Untuk menyimpan pengaturan Anda dan terus mengerjakannya nanti, pilih OK untuk menutup dialog sistem Properti Layanan EC2. Jika Anda telah selesai menyesuaikan instans Anda dan ingin membuat AMI dari instans itu, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.

## Memecahkan masalah dengan agen peluncuran EC2config

Informasi berikut dapat membantu Anda memecahkan masalah dengan layanan EC2Config.

### Perbarui EC2Config pada instans yang tidak dapat dijangkau

Gunakan prosedur berikut untuk memperbarui layanan EC2Config pada instans Windows Server yang tidak dapat diakses menggunakan Desktop Jarak Jauh.

Untuk memperbarui EC2config pada instans Amazon EBS-backed Windows yang tidak dapat Anda sambungkan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Cari instans yang terpengaruh. Pilih instans dan pilih status Instans, lalu pilih Hentikan instans.

   > **Warning**
   >
   > Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

4. Pilih Luncurkan instans dan buat instans `t2.micro` sementara di Zona Ketersediaan yang sama dengan instans yang terpengaruh. Gunakan AMI yang berbeda dari AMI yang Anda gunakan untuk meluncurkan instans yang terpengaruh.

   > **Important**
   >
   > Jika Anda tidak membuat instans di Zona Ketersediaan yang sama dengan instans yang terpengaruh, Anda tidak akan dapat melampirkan volume root dari instans yang terpengaruh ke instans baru.

5. Pada konsol EC2 pilih Volume.
6. Cari volume root dari instans yang terdampak. Lepaskan volume dan Lampirkan volume ke instans sementara yang Anda buat sebelumnya. Lampirkan dengan nama perangkat default (xvdf).
7. Gunakan Remote Desktop untuk terhubung ke instans sementara, dan kemudian gunakan utilitas Disk Management agar volume tersedia untuk digunakan.
8. Unduh versi terbaru layanan EC2Config. Ekstrak file dari `.zip` file ke`Temp` direktori pada drive yang Anda lampirkan.
9. Pada instans sementara, buka kotak dialog Run (Jalankan), ketik, **`regedit`**dan tekan Enter.

10. Pilih HKEY_LOCAL_MACHINE. Dari menu File, pilih Muat Hive. Pilih drive dan kemudian arahkan ke dan buka file berikut: Windows\System32\config\SOFTWARE. Saat diminta, tentukan nama kunci.
11. Pilih kunci yang baru saja Anda muat dan arahkan ke Microsoft\Windows \CurrentVersion. Memilih kunci RunOnce. Jika kunci ini tidak ada, pilih CurrentVersion dari menu konteks (klik kanan), pilih Baru lalu pilih Kunci. Beri nama kunci RunOnce.
12. Dari menu konteks (klik kanan), pilih kunci RunOnce, lalu pilih Baru, lalu pilih Nilai String. Masukkan Ec2Install sebagai nama dan C:\Temp\Ec2Install.exe /quiet sebagai data.
13. MemilihHKEY_LOCAL_MACHINE*specified key name*\Microsoft\Windows NT \CurrentVersion\Winlogon kunci. Dari menu konteks (klik kanan) pilih Baru, lalu pilih Nilai String. Masukkan **AutoAdminLogon** sebagai nama dan **1** sebagai data nilai.
14. Memilih kunci HKEY_LOCAL_MACHINE*specified key name*\Microsoft\Windows NT \CurrentVersion\Winlogon>. Dari menu konteks (klik kanan) pilih Baru, lalu pilih Nilai String. Masukkan **DefaultUserName** sebagai nama dan **Administrator** sebagai data nilai.
15. Memilih kunci HKEY_LOCAL_MACHINE*specified key name*\Microsoft\Windows NT \CurrentVersion\Winlogon. Dari menu konteks (klik kanan) pilih Baru, lalu pilih Nilai String. Ketik **DefaultPassword** sebagai nama dan masukkan kata sandi di data nilai.
16. Di panel navigasi Editor Registri, pilih kunci sementara yang Anda buat saat pertama kali membuka Editor Registri.
17. Dari File pilihan, pilih Pembongkaran Hive.
18. Di Pemanfaatan Manajemen DiskIT, pilih drive yang Anda lampirkan sebelumnya, buka menu konteks (klik kanan), dan pilih Offline.
19. Di konsol Amazon EC2, lepaskan volume yang terpengaruh dari instans sementara dan pasang kembali ke instans Anda dengan nama perangkat /dev/sda1. Anda harus menentukan nama perangkat ini untuk menetapkan volume sebagai volume root.
20. Menghentikan dan memulai instans Amazon EC2 instans.
21. Setelah instans dimulai, periksa log sistem dan pastikan bahwa Anda melihat pesan Windows siap digunakan.
22. Buka Penyunting Registri dan pilih HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon. Hapus kunci Nilai String yang Anda buat sebelumnya: AutoAdminLogon, DefaultUserName, dan DefaultPassword.
23. Hapus atau hentikan instans sementara yang Anda buat dalam prosedur ini.

## Riwayat versi EC2Config

Tabel berikut menjelaskan versi EC2Config yang dirilis. Untuk informasi tentang pembaruan untuk SSM Agent, lihat Catatan Rilis Systems Manager SSM Agent.

> **Important**
>
> EC2config telah mencapai akhir dukungan. Hanya versi terbaru dari agen EC2config yang tersedia untuk diunduh. Versi sebelumnya ditandai sebagai pribadi.

| Versi | Detail | Tanggal rilis |
|---|---|---|
| 4.9.5777 | • Memperbaiki masalah ketika konfigurasi RSS disetel secara tidak benar untuk beberapa jenis instance.<br>• Versi baru SSM Agent `3.3.484.0` . | 17 Juni 2024 |
| 4.9.5554 | • Membatasi devolusi nama domain berdasarkan entri registri: `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Dnscache\Parameters\DomainNameDevolutionLevel` .<br>• Versi baru SSM Agent `3.2.1630.0` . | 4 Oktober 2023 |
| 4.9.5467 | • Menambahkan kemampuan coba lagi untuk menemukan port konsol.<br>• Versi baru SSM Agent `3.1.2282.0` . | 1 Agustus 2023 |
| 4.9.5288 | • AWS Core SDK yang diperbarui ke versi `3.7.103.23` .<br>• Memperbaiki masalah saat dokumen `AWS-UpdateEC2Config` SSM gagal diperbarui `EC2Config` pada instans yang diaktifkan hanya dengan IMDSv2.<br>• | 8 Maret 2023 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| | Versi baru SSM Agent `3.1.2144.0` . | |
| 4.9.5231 | • Versi baru SSM Agent 3.1.1927.0. | 14 Februari 2023 |
| 4.9.5103 | • Memperbaiki masalah di mana volume fana diidentifikasi secara tidak benar pada keluarga instans r5d dan i4i.<br>• Versi baru SSM Agent 3.1.1856.0. | 5 Desember 2022 |
| 4.9.5064 | • Diperbarui untuk menggunakan informasi segmen PCI guna memilih port konsol.<br>• PowerShell Skrip yang ditandatangani dan menambahkan header hak cipta.<br>• Logika pemilihan adaptor jaringan utama tetap.<br>• Versi baru SSM Agent 3.1.1732.0. | 16 November 2022 |
| 4.9.4588 | • Logika tunggu IMDS yang diperbarui untuk membuat permintaan IMDSv2 saja.<br>• Menambahkan pustaka bersama agen peluncuran libec2launch.dll.<br>• Versi baru SSM Agent 3.1.1188.0. | 31 Mei 2022 |
| 4.9.4556 | • Menambahkan logika tunggu untuk memastikan inisialisasi penuh NIC sebelum digunakan.<br>• Versi baru Log4Net 2.0.14.0 mengambil patch keamanan.<br>• Versi baru SSM Agent 3.1.1045.0 mengambil patch keamanan. | 1 Maret 2022 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.9.4536 | • Memperbaiki masalah saat data pengguna mogok saat folder Temp hilang.<br>• Versi baru SSM Agent 3.1.804.0. | 31 Januari 2022 |
| 4.9.4508 | • Memperbaiki masalah untuk menghitung jalur skrip diskpart dengan benar.<br>• Versi baru SSM Agent 3.1.338.0. | 6 Oktober 2021 |
| 4.9.4500 | • `Install-EgpuManagerConfig` yang diperbarui dengan dukungan IMDS v2.<br>• Tautan web yang diperbarui untuk menggunakan https.<br>• Versi baru SSM Agent 3.1.282.0 | 7 September 2021 |
| 4.9.4419 | • Memperbaiki logika fallback IMDS versi 1<br>• Memperbarui semua penggunaan direktori temp Windows untuk direktori temp EC2Config<br>• Versi baru SSM Agent 3.0.1124.0 | 2 Juni 2021 |
| 4.9.4381 | • Menambahkan dukungan untuk skema dokumen SSM versi 2.2 di EC2ConfigUpdater<br>• Menambahkan versi paket AWS Nitro Enclaves ke log konsol<br>• Versi baru SSM Agent 3.0.529.0 | 4 Mei 2021 |
| 4.9.4326 | • Menghapus semua tautan di pengaturan UI<br>• Ini adalah versi EC2Config terakhir yang mendukung Windows Server 2008. | 3 Maret 2021 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.9.4279 | • Memperbaiki masalah keamanan yang terkait dengan tugas terjadwal `Ec2ConfigMonitor`<br>• Memperbaiki masalah pemetaan huruf drive dan jumlah disk ephemeral yang salah<br>• Menambahkan `OsCurrentBuild` dan `OsReleaseId` ke output konsol<br>• Versi baru SSM Agent 2.3.871.0 | 11 Desember 2020 |
| 4.9.4222 | • Memperbaiki logika fallback IMDS versi 1<br>• Versi baru SSM Agent 2.3.842.0 | 7 April 2020 |
| 4.9.4122 | • Menambahkan dukungan untuk IMDS V2<br>• Versi baru SSM Agent 2.3.814.0 | 4 Maret 2020 |
| 4.9.3865 | • Memperbaiki masalah pendeteksian port COM untuk Windows Server 2008 R2 pada instans metal<br>• Versi baru SSM Agent 2.3.722.0 | 31 Oktober 2019 |
| 4.9.3519 | • Versi baru SSM Agent 2.3.634.0 | 18 Juni 2019 |
| 4.9.3429 | • Versi baru SSM Agent 2.3.542.0 | 25 April 2019 |
| 4.9.3289 | • Versi baru 2.3.444.0 | 11 Februari 2019 |
| 4.9.3270 | • Menambahkan plugin untuk mengatur monitor agar tidak pernah mati untuk memperbaiki masalah ACPI<br>• Edisi dan versi SQL Server yang ditulis ke konsol<br>• Versi baru SSM Agent 2.3.415.0 | 22 Januari 2019 |
| 4.9.3230 | • Deskripsi Pemetaan Huruf Drive yang diperbarui agar lebih selaras dengan fungsionalitas<br>• Versi baru SSM Agent 2.3.372.0 | 10 Januari 2019 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.9.3160 | • Peningkatan waktu tunggu untuk NIC utama<br>• Menambahkan konfigurasi default untuk pengaturan RSS dan Menerima Antrean untuk perangkat ENA<br>• Hibernasi dinonaktifkan selama Sysprep<br>• Versi baru SSM Agent 2.3.344.0<br>• AWS SDK yang ditingkatkan ke 3.3.29.13 | 15 Desember 2018 |
| 4.9.3067 | • Perbaikan yang dilakukan pada hibernasi instans<br>• Versi baru SSM Agent 2.3.235.0 | 8 November 2018 |
| 4.9.3034 | • Ditambahkan rute 169.254.169. 253/32 untuk server DNS<br>• Versi baru SSM Agent 2.3.193.0 | 24 Oktober 2018 |
| 4.9.2986 | • Menambahkan penandatanganan untuk semua biner terkait EC2Config<br>• Versi baru SSM Agent 2.3.136.0 | 11 Oktober 2018 |
| 4.9.2953 | Versi baru SSM Agent (2.3.117.0) | 2 Oktober 2018 |
| 4.9.2926 | Versi baru SSM Agent (2.3.68.0) | 18 September 2018 |
| 4.9.2905 | • Versi baru SSM Agent (2.3.50.0)<br>• Ditambahkan rute 169.254.169. 123/32 ke layanan waktu AMZN<br>• Ditambahkan rute 169.254.169. 249/32 ke layanan lisensi GRID<br>• Memperbaiki masalah yang menyebabkan volume EBS NVMe untuk ditandai sebagai ephemeral | 17 September 2018 |
| 4.9.2854 | Versi baru SSM Agent (2.3.13.0) | 17 Agustus 2018 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.9.2831 | Versi baru SSM Agent (2.2.916.0) | 7 Agustus 2018 |
| 4.9.2818 | Versi baru SSM Agent (2.2.902.0) | 31 Juli 2018 |
| 4.9.2756 | Versi baru SSM Agent (2.2.800.0) | 27 Juni 2018 |
| 4.9.2688 | Versi baru SSM Agent (2.2.607.0) | 25 Mei 2018 |
| 4.9.2660 | Versi baru SSM Agent (2.2.546.0) | 11 Mei 2018 |
| 4.9.2644 | Versi baru SSM Agent (2.2.493.0) | 26 April 2018 |
| 4.9.2586 | Versi baru SSM Agent (2.2.392.0) | 28 Maret 2018 |
| 4.9.2565 | • Versi baru SSM Agent (2.2.355.0)<br>• Memperbaiki masalah pada instans M5 dan C5 (tidak dapat menemukan driver PV)<br>• Tambahkan logging konsol untuk tipe instans, driver PV terbaru, dan driver NVMe | 13 Maret 2018 |
| 4.9.2549 | Versi baru SSM Agent (2.2.325.0) | 8 Maret 2018 |
| 4.9.2461 | Versi baru SSM Agent (2.2.257.0) | 15 Februari 2018 |
| 4.9.2439 | Versi baru SSM Agent (2.2.191.0) | 6 Februari 2018 |
| 4.9.2400 | Versi baru SSM Agent (2.2.160.0) | 16 Januari 2018 |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.9.2327 | • Versi baru SSM Agent (2.2.120.0)<br>• Penambahan penemuan port COM pada instans bare metal Amazon EC2<br>• Menambahkan Hyper-V pencatatan status pada instans bare metal Amazon EC2 | 2 Januari 2018 |
| 4.9.2294 | Versi baru SSM Agent (2.2.103.0) | 4 Desember 2017 |
| 4.9.2262 | Versi baru SSM Agent (2.2.93.0) | 15 November 2017 |
| 4.9.2246 | Versi baru SSM Agent (2.2.82.0) | 11 November 2017 |
| 4.9.2218 | Versi baru SSM Agent (2.2.64.0) | 29 Oktober 2017 |
| 4.9.2212 | Versi baru SSM Agent (2.2.58.0) | 23 Oktober 2017 |
| 4.9.2203 | Versi baru SSM Agent (2.2.45.0) | 19 Oktober 2017 |
| 4.9.2188 | Versi baru SSM Agent (2.2.30.0) | 10 Oktober 2017 |
| 4.9.2180 | • Versi baru SSM Agent (2.2.24.0)<br>• Menambahkan plugin Elastic GPU untuk instans GPU | 5 Oktober 2017 |
| 4.9.2143 | Versi baru SSM Agent (2.2.16.0) | 1 Oktober 2017 |
| 4.9.2140 | Versi baru SSM Agent (2.1.10.0) | |
| 4.9.2130 | Versi baru SSM Agent (2.1.4.0) | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.9.2106 | Versi baru SSM Agent (2.0.952.0) | |
| 4.9.2061 | Versi baru SSM Agent (2.0.922.0) | |
| 4.9.2047 | Versi baru SSM Agent (2.0.913.0) | |
| 4.9.2031 | Versi baru SSM Agent (2.0.902.0) | |
| 4.9.2016 | • Versi baru SSM Agent (2.0.879.0)<br>• Memperbaiki jalur direktori CloudWatch Logs untuk Windows Server 2003 | |
| 4.9.1981 | • Versi baru SSM Agent (2.0.847.0)<br>• Memperbaiki masalah dengan pembuatan `important.txt` sedang di volume EBS. | |
| 4.9.1964 | Versi baru SSM Agent (2.0.842.0) | |
| 4.9.1951 | • Versi baru SSM Agent (2.0.834.0)<br>• Memperbaiki masalah huruf drive yang tidak dipetakan dari Z: untuk drive singkat. | |
| 4.9.1925 | • Versi baru SSM Agent (2.0.822.0)<br>• [Bug] Versi ini bukan target pembaruan yang valid dari SSM Agent v4.9.1775. | |
| 4.9.1900 | Versi baru SSM Agent (2.0.805.0) | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.9.1876 | • Versi baru SSM Agent (2.0.796.0)<br>• Memperbaiki masalah dengan output/error pengalihan untuk eksekusi data pengguna admin. | |
| 4.9.1863 | • Versi baru SSM Agent (2.0.790.0)<br>• Memperbaiki masalah terkait pemasangan beberapa volume EBS ke instans Amazon EC2.<br>• Dit CloudWatch ingkatkan untuk mengambil jalur konfigurasi, menjaga kompatibilitas mundur. | |
| 4.9.1791 | Versi baru SSM Agent (2.0.767.0) | |
| 4.9.1775 | Versi baru SSM Agent (2.0.761.0) | |
| 4.9.1752 | Versi baru SSM Agent (2.0.755.0) | |
| 4.9.1711 | Versi baru SSM Agent (2.0.730.0) | |
| 4.8.1676 | Versi baru SSM Agent (2.0.716.0) | |
| 4.7.1631 | Versi baru SSM Agent (2.0.682.0) | |
| 4.6.1579 | • Versi baru SSM Agent (2.0.672.0)<br>• Memperbaiki masalah pembaruan agen dengan v4.3, v4.4, dan v4.5 | |
| 4.5.1534 | Versi baru SSM Agent (2.0.645.1) | |
| 4.4.1503 | Versi baru SSM Agent (2.0.633.0) | |
| 4.3.1472 | Versi baru SSM Agent (2.0.617.1) | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 4.2.1442 | Versi baru SSM Agent (2.0.599.0) | |
| 4.1.1378 | Versi baru SSM Agent (2.0.558.0) | |
| 4.0.1343 | • Run Command, State Manager, CloudWatch agen, dan dukungan bergabung domain telah dipindahkan ke agen lain yang disebut Agen SSM. SSM Agent akan diinstal sebagai bagian dari pemutakhiran EC2Config. Untuk informasi selengkapnya, lihat EC2config dan AWS Systems Manager.<br>• Jika Anda memiliki proxy yang disiapkan di EC2Config, maka Anda perlu memperbarui pengaturan proxy untuk SSM Agent sebelum melakukan upgrade. Jika Anda tidak memperbarui pengaturan proxy, Anda tidak akan dapat menggunakan Jalankan Perintah untuk mengelola instans Anda. Untuk menghindari hal ini, lihat informasi berikut sebelum memperbarui ke versi yang lebih baru: Menginstal dan Mengonfigurasi SSM Agent pada Instans Windows di Panduan Pengguna AWS Systems Manager .<br>• Jika sebelumnya Anda mengaktifkan CloudWatch integrasi pada instans Anda dengan menggunakan file konfigurasi lokal (`AWS.EC2.Windows.CloudWatch.json` ), Anda harus mengonfigurasi file untuk bekerja dengan Agen SSM. | |
| 3.19.1153 | • Re-enabled plugin aktivasi untuk contoh dengan AWS KMS konfigurasi lama. Lewati aktivasi untuk pengguna BYOL.<br>• Ubah perilaku TRIM default untuk dinonaktifkan selama operasi format disk dan menambahkan FormatWith TRIM untuk mengganti InitializeDisks plugin dengan data pengguna. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 3.18.1118 | • Perbaiki untuk menambahkan rute dengan andal ke adaptor jaringan utama.<br>• Pembaruan untuk meningkatkan dukungan untuk AWS layanan. | |
| 3.17.1032 | • Memperbaiki log sistem duplikat yang muncul saat filter disetel ke kategori yang sama.<br>• Perbaikan untuk mencegah hang selama inisialisasi disk. | |
| 3.16.930 | Menambahkan dukungan ke log peristiwa "Jendela Siap digunakan" ke Log Peristiwa Windows saat dimulai. | |
| 3.15.880 | Perbaikan untuk mengizinkan pengunggahan keluaran System Manager Run Command ke nama bucket S3 dengan '.' karakter. | |
| 3.14.786 | Menambahkan dukungan untuk mengganti pengaturan Initializ eDisks plugin. Contoh: Untuk mempercepat inisialisasi disk SSD, Anda dapat menonaktifkan TRIM untuk sementara dengan menentukan ini di userdata:<br>< InitializeDrivesSettings ><> POTONG <SettingsGroup /> <FormatWithout/SettingsGroupInitializeDrivesSettings | |
| 3.13.727 | System Manager Run Command - Perbaikan untuk memproses perintah dengan andal setelah windows reboot. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 3.12.649 | • Perbaiki untuk menangani reboot dengan anggun saat berjalancommands/scripts.<br>• Perbaiki untuk membatalkan perintah yang berjalan dengan andal.<br>• Tambahkan dukungan untuk (opsional) mengunggah log MSI ke S3 saat menginstal aplikasi melalui Systems Manager Run Command. | |
| 3.11.521 | • Perbaikan untuk mengaktifkan pembuatan sidik jari RDP untuk Windows Server 2003.<br>• Perbaikan untuk menyertakan perbedaan zona waktu dan UTC di baris log EC2Config.<br>• Dukungan Systems Manager untuk menjalankan perintah Run Command secara paralel.<br>• Kembalikan perubahan sebelumnya untuk menghadirkan disk yang dipartisi secara online. | |
| 3.10.442 | • Memperbaiki kegagalan konfigurasi Systems Manager saat menginstal aplikasi MSI.<br>• Perbaiki untuk menghadirkan disk penyimpanan online dengan andal.<br>• Pembaruan untuk meningkatkan dukungan untuk AWS layanan. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 3.9.359 | • Perbaiki dalam skrip pasca Sysprep untuk membiarkan konfigurasi pembaruan windows dalam status default.<br>• Perbaiki plugin pembuat kata sandi untuk meningkatkan keandalan dalam mendapatkan pengaturan kebijakan kata sandi GPO.<br>• Bat EC2Config/SSM asi izin folder log ke grup Administrator lokal.<br>• Pembaruan untuk meningkatkan dukungan untuk AWS layanan. | |
| 3.8.294 | • Memperbaiki masalah CloudWatch yang mencegah log diunggah saat tidak berada di drive utama.<br>• Meningkatkan proses inisialisasi disk dengan menambahkan logika coba lagi.<br>• Menambahkan penanganan kesalahan yang ditingkatkan ketika SetPassword plugin kadang-kadang gagal selama pembuatan AMI.<br>• Pembaruan untuk meningkatkan dukungan untuk AWS layanan. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 3.7.308 | • Perbaikan pada utilitas ec2config-cli untuk pengujian konfigurasi dan pemecahan masalah dalam instans.<br>• Hindari menambahkan rute statis untuk AWS KMS dan layanan meta-data pada adaptor OpenVPN.<br>• Memperbaiki masalah ketika eksekusi data pengguna tidak mematuhi tag "persisten".<br>• Penanganan kesalahan yang lebih baik saat masuk ke konsol EC2 tidak tersedia.<br>• Pembaruan untuk meningkatkan dukungan untuk AWS layanan. | |
| 3.6.269 | • Perbaikan keandalan aktivasi Windows untuk pertama kali menggunakan alamat lokal tautan 169.254.0. 250/251 untuk mengaktifkan windows melalui AWS KMS<br>• Penanganan proxy yang lebih baik untuk skenario Systems Manager, Aktivasi Windows, dan Penggabungan Domain<br>• Memperbaiki masalah di mana baris duplikat akun pengguna ditambahkan ke file jawaban Sysprep | |
| 3.5.228 | • Mengatasi skenario di mana CloudWatch plugin mungkin menggunakan CPU dan memori yang berlebihan untuk membaca Log Peristiwa Windows<br>• Menambahkan tautan ke dokumentasi CloudWatch konfigurasi di UI Pengaturan EC2Config | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 3.4.212 | • Perbaikan untuk EC2config saat digunakan dalam kombinasi dengan. VM-Import<br>• Memperbaiki masalah penamaan layanan di pemasang WiX. | |
| 3.3.174 | • Penanganan pengecualian yang lebih baik untuk Systems Manager dan kegagalan penggabungan domain.<br>• Ubah untuk mendukung versioning skema SSM Systems Manager.<br>• Memperbaiki format disk sementara pada Win2K3.<br>• Ubah untuk mendukung konfigurasi ukuran disk yang lebih besar dari 2TB.<br>• Mengurangi penggunaan memori virtual dengan menyetel mode GC ke default.<br>• Dukungan untuk mengunduh artefak dari jalur UNC di plugin `aws:psModule` dan `aws:application` .<br>• Peningkatan logging untuk plugin aktivasi Windows. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 3.2.97 | • Peningkatan performa dengan penundaan pemuatan rakitan Systems Manager SSM.<br>• Penanganan pengecualian yang lebih baik untuk format sysprep2008.xml yang salah.<br>• Dukungan baris perintah untuk konfigurasi "Terapkan" Systems Manager.<br>• Ubah untuk mendukung penggabungan domain ketika ada penggantian nama komputer yang tertunda.<br>• Dukungan untuk parameter opsional di plugin `aws:applications` .<br>• Dukungan untuk array perintah di plugin `aws:psModule` . | |
| 3.0.54 | • Aktifkan dukungan untuk Systems Manager.<br>• Secara otomatis domain bergabung dengan instance EC2 Windows ke AWS direktori melalui Manajer Sistem.<br>• Konfigurasikan dan unggah CloudWatch logs/metrics melalui Manajer Sistem.<br>• Inst PowerShell al modul melalui Manajer Sistem.<br>• Instal aplikasi MSI melalui System Manager. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.4.233 | • Menambahkan tugas terjadwal untuk memulihkan EC2Config dari kegagalan startup layanan.<br>• Perbaikan pesan kesalahan log Konsol.<br>• Pembaruan untuk meningkatkan dukungan untuk AWS layanan. | |
| 2.3313 | • Memperbaiki masalah dengan konsumsi memori yang besar dalam beberapa kasus ketika fitur CloudWatch Log diaktifkan.<br>• Memperbaiki bug pemutakhiran sehingga ec2config versi yang lebih rendah dari 2.1.19 sekarang dapat dimutakhirkan ke versi terbaru.<br>• Pengecualian pembukaan port COM yang diperbarui agar lebih ramah pengguna dan berguna dalam log.<br>• Ec2configServiceSettings UI menonaktifkan pengubahan ukuran dan memperbaiki penempatan tampilan atribusi dan versi di UI. | |
| 2.2.12 | • Ditangani NullPointerException saat menanyakan kunci registri untuk menentukan status Windows Sysprep yang kadang-kadang mengembalikan null.<br>• Membebaskan sumber daya yang tidak terkelola pada akhirnya diblokir. | |
| 2.2.11 | Memperbaiki masalah di CloudWatch plugin untuk menangani baris log kosong. | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.10 | • Menghapus konfigurasi pengaturan CloudWatch Log melalui UI.<br>• Aktifkan pengguna untuk menentukan pengaturan CloudWatch Log dalam `%ProgramFiles%\Amazon\Ec2ConfigService\Settings\AWS.EC2.Windows.CloudWatch.json` file untuk memungkinkan penyempurnaan di masa mendatang. | |
| 2.2.9 | Memperbaiki pengecualian yang tidak tertangani dan menambahkan logging. | |
| 2.2.8 | • Memperbaiki pemeriksaan versi Windows OS in EC2Config Installer untuk mendukung Windows Server 2003 SP1 dan setelahnya.<br>• Memperbaiki penanganan nilai null saat membaca kunci registri yang terkait dengan memperbarui file konfigurasi Sysprep. | |
| 2.2.7 | • Menambahkan dukungan untuk EC2Config agar berjalan selama eksekusi Sysprep untuk Windows 2008 dan yang lebih tinggi.<br>• Penanganan pengecualian dan logging yang lebih baik untuk diagnostik yang lebih baik | |
| 2.2.6 | • Mengurangi beban pada instance dan pada CloudWatch Log saat mengunggah peristiwa log.<br>• Mengatasi masalah peningkatan di mana plug-in CloudWatch Logs tidak selalu tetap diaktifkan | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.5 | • Menambahkan dukungan untuk mengunggah CloudWatch log ke Layanan Log.<br>• Memperbaiki masalah kondisi balapan di Ec2OutputRDPCert pengaya<br>• Mengubah opsi pemulihan Layanan EC2config untuk Restart dari TakeNoAction<br>• Menambahkan lebih banyak informasi pengecualian saat EC2Config Rusak | |
| 2.2.4 | • Memperbaiki kesalahan ketik di PostSysprep.cmd<br>• Memperbaiki bug yang tidak disematkan otomatis oleh EC2Config ke menu awal untuk OS2012 + | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.3 | • Menambahkan opsi untuk menginstal EC2Config tanpa langsung memulai layanan setelah instalasi. Untuk menggunakan, jalankan 'Ec2Install.exestart=false' dari command prompt<br>• Ditambahkan parameter di plugin wallpaper untuk mengontro l adding/removing wallpaper. Untuk menggunakan, jalankan 'Ec2WallpaperInfo.exeset' atau 'returnEc2WallpaperInfo.exe ' dari command prompt<br>• Menambahkan pemeriksaan RealTimeIsUniversal kunci, menghasilkan pengaturan yang salah dari kunci RealTimeI sUniveral registri ke Konsol<br>• Menghapus ketergantungan EC2Config pada folder temp Windows<br>• Dih UserData apus ketergantungan eksekusi pada.Net 3.5 | |
| 2.2.2 | • Menambahkan pemeriksaan ke perilaku penghentian layanan untuk memastikan bahwa sumber daya sedang dirilis<br>• Memperbaiki masalah waktu eksekusi yang lama saat bergabung dengan domain | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.2.1 | • Penginstal yang diperbarui untuk memungkinkan pemutakhiran dari versi sebelumnya<br>• Memper Ec2WallpaperInfo baiki bug di. Net4.5 hanya lingkungan<br>• Memperbaiki bug deteksi driver yang terputus-putus<br>• Menambahkan opsi instal diam. Jalan Ec2Install.exe kan dengan opsi '-q '. misalnya:' -q' Ec2Install.exe | |
| 2.2.0 | • Menambahkan dukungan untuk.Net4 dan. Net4.5 hanya lingkungan<br>• Penginstal yang Diperbarui | |
| 2.1.19 | • Menambahkan dukungan pelabelan disk sementara saat menggunakan driver jaringan Intel (mis. Tipe instans C3). Untuk informasi selengkapnya, lihat Jaringan yang disempurnakan di instans Amazon EC2.<br>• Menambahkan Versi Asli AMI dan dukungan Nama Asli AMI ke output konsol<br>• Membuat perubahan pada Output Konsol agar konsisten formatting/parsing<br>• File Bantuan yang Diperbarui | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.18 | • Menambahkan Objek WMI EC2config untuk pemberitahuan Penyelesaian (-Namespace root\ Amazon -Class EC2_) ConfigService<br>• Kueri Peningkatan Performa WMI Startup dengan Log Peristiwa besar; dapat menyebabkan CPU tinggi yang berkepanjangan selama eksekusi awal | |
| 2.1.17 | • Memperbaiki masalah UserData eksekusi dengan pengisian buffer Output Standar dan Kesalahan Standar<br>• Sidik jari RDP yang salah yang terkadang muncul di Output Konsol untuk> = w2k8 OS telah diperbaiki<br>• Output Konsol sekarang berisi 'RDPCERTIFICATE-Su bjectName:' untuk Windows 2008+, yang berisi nilai nama mesin<br>• Menambahkan D:\ ke menu tarik-turun Pemetaan Huruf Drive<br>• Tombol Bantuan dipindahkan ke kanan atas dan diubah look/ feel<br>• Menambahkan tautan survei Umpan Balik di kanan atas | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.16 | • Tab Umum menyertakan tautan ke halaman unduh EC2Config untuk Versi baru<br>• Hamparan Wallpaper Desktop sekarang disimpan di folder Appdata Lokal Pengguna alih-alih Dokumen Saya untuk mendukung MyDoc pengalihan<br>• Nama MSSQLServer disinkronkan dengan sistem dalam skrip (2008+) Post-Sysprep<br>• Folder Aplikasi yang Diurutkan Ulang (memindahkan file ke direktori Plugin dan menghapus file duplikat)<br>• Output Log Sistem Diubah (Konsol):<br>• * Telah memindahkan ke format tanggal, nama, nilai untuk penguraian yang lebih mudah (Harap mulai memigrasikan dependensi ke format baru)<br>• * Ditambahkan status plugin Ec2SetPassword "<br>• * Telah menambahkan waktu Mulai dan Akhir Sysprep<br>• Memperbaiki masalah Ephemeral Disks yang tidak diberi label sebagai 'Penyimpanan Sementara' untuk Sistem Operasi non-Inggris<br>• Memperbaiki gagal Uninstall EC2Config setelah menjalankan Sysprep | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.15 | • Permintaan yang dioptimalkan ke layanan Metadata<br>• Metadata sekarang melewati Pengaturan Proxy<br>• Disk Ephemeral diberi label sebagai 'Penyimpanan Temporar' dan Important.txt ditempatkan pada volume saat ditemukan (hanya driver Citrix PV). Untuk informasi selengkapnya, lihat Tingkatkan driver PV pada instans Windows EC2.<br>• Ephemeral Disk menetapkan huruf drive dari Z ke A (hanya driver Citrix PV) - penetapan dapat ditimpa menggunakan plugin Pemetaan Huruf Drive dengan label Volume 'Penyimpanan Sementara X' di mana x adalah angka 0-25)<br>• UserData sekarang berjalan segera setelah 'Windows Siap' | |
| 2.1.14 | Perbaikan wallpaper desktop | |
| 2.1.13 | • Wallpaper desktop akan menampilkan nama host secara default<br>• Ketergantungan yang dihapus pada layanan Windows Time<br>• Rute ditambahkan jika ada beberapa IP ditugaskan ke satu antarmuka | |
| 2.111 | • Perubahan dilakukan pada Plugin Ec2Activation<br>• -Memverifikasi status Aktivasi setiap 30 hari<br>• -Jika Masa Tenggang memiliki sisa 90 hari (dari 180 hari), coba kembali aktivasi | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.10 | • Hamparan wallpaper desktop tidak lagi dipertahankan dengan Sysprep atau Mati tanpa Sysprep<br>• Opsi data pengguna untuk dijalankan pada setiap layanan dimulai dengan <persist>true</persist><br>• Mengubah lokasi dan nama/DisableWinUpdate.cmd menjadi/Scripts/PostSysprep.cmd<br>• Kata sandi administrator diatur untuk tidak kedaluwarsa secara default di/Scripts/PostSysprep.cmd<br>• Uninstall akan menghapus PostSysprep skrip EC2config dari c:\windows\setup\script\ CommandComplete.cmd<br>• Tambah Rute mendukung metrik antarmuka kustom | |
| 2.1.9 | UserData Eksekusi tidak lagi terbatas pada 3851 Karakter | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.7 | • Versi OS dan pengenal bahasa yang ditulis pada konsol<br>• Versi EC2Config yang ditulis pada konsol<br>• Versi driver PV yang ditulis pada konsol<br>• Deteksi Pemeriksaan Bug dan output ke konsol pada boot berikutnya ketika ditemukan<br>• Opsi ditambahkan ke config.xml untuk mempertahankan kredensial Sysprep<br>• Tambahkan logika Coba Ulang Rute jika ENI tidak tersedia di awal<br>• PID eksekusi Data Pengguna ditulis ke konsol<br>• Panjang kata sandi minimum yang dihasilkan diambil dari GPO<br>• Setel layanan mulai mencoba lagi 3 kali<br>• Menambahkan contoh S3_ DownloadFile.ps1 dan S3_Upload file.ps1 ke folder /Scripts | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.6 | • Informasi versi ditambahkan ke tab Umum<br>• Mengganti nama tab Bundel menjadi Gambar<br>• Menyederhanakan proses menentukan kata sandi dan memindahkan UI terkait kata sandi dari tab Umum ke tab Gambar<br>• Mengganti nama tab Pengaturan Disk menjadi Penyimpanan<br>• Menambahkan tab Dukungan dengan alat umum untuk pemecahan masalah<br>• `sysprep.ini` Windows Server 2003 diatur untuk memperluas partisi OS secara default<br>• Menambahkan alamat IP privat ke wallpaper<br>• Alamat IP pribadi ditampilkan di wallpaper<br>• Menambahkan logika coba lagi untuk output Konsol<br>• Memperbaiki pengecualian Com port untuk aksesibilitas metadata – menyebabkan EC2Config berakhir sebelum output konsol ditampilkan<br>• Memeriksa status aktivasi pada setiap boot -- aktifkan seperlunya<br>• Memperbaiki masalah jalur relatif - disebabkan saat menjalank an pintasan wallpaper secara manual dari folder startup; menunjuk ke Administrator/logs<br>• Memperbaiki warna latar belakang default untuk pengguna Windows Server 2003 (selain Administrator) | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.2 | • Stempel waktu konsol dalam UTC (Zulu)<br>• Tampilan hyperlink pada tab Sysprep dihapus<br>• Penambahan fitur untuk memperluas Volume Root secara dinamis saat boot pertama untuk Windows 2008+<br>• Ketika Set-Password diaktifkan, sekarang secara otomatis mengaktifkan EC2config untuk mengatur kata sandi<br>• EC2Config memeriksa status aktivasi sebelum menjalankan Sysprep (menampilkan peringatan jika tidak diaktifkan)<br>• `Sysprep.xml` Windows Server 2003 sekarang default ke zona waktu UTC, bukan Pasifik<br>• Server Aktivasi Acak<br>• Mengganti nama tab Pemetaan Drive menjadi Pengaturan Disk<br>• Pindah Inisialisasi item UI Drive dari Umum ke tab Pengaturan Disk<br>• Tombol bantuan sekarang mengarah ke file bantuan HTML<br>• File bantuan HTML yang diperbarui dengan perubahan<br>• ‘Catatan’ yang diperbarui untuk Pemetaan Huruf Drive<br>• Ditambahkan InstallUpdates.ps1 ke folder /Scripts untuk mengotomatiskan Patch dan pembersihan sebelum Sysprep | |

| Versi | Detail | Tanggal rilis |
| --- | --- | --- |
| 2.1.0 | • Wallpaper desktop menampilkan informasi instance secara default saat masuk pertama (tidak disconnect/reconnect)<br>• PowerShell dapat dijalankan dari data pengguna dengan mengelilingi kode dengan<powershell></powershell> | |

## Gunakan EC2 Fast Launch untuk instans Windows Anda

Saat Anda mengonfigurasi AMI Windows Server untuk Peluncuran Cepat EC2, Amazon EC2 membuat serangkaian snapshot yang telah disediakan sebelumnya untuk digunakan untuk peluncuran lebih cepat, sebagai berikut.

1. Amazon EC2 meluncurkan satu set instans t3 temporer, berdasarkan pengaturan Anda.
2. Saat setiap instans temporer menyelesaikan langkah peluncuran standar, Amazon EC2 membuat snapshot instans yang telah disediakan sebelumnya. Poses ini menyimpan snapshot di bucket Amazon S3.
3. Saat snapshot sudah siap, Amazon EC2 mengakhiri instans t3 terkait untuk menekan biaya sumber daya.
4. Lain kali Amazon EC2 meluncurkan instance dari EC2 Fast Launch mengaktifkan AMI, ia menggunakan salah satu snapshot untuk secara signifikan mengurangi waktu yang diperlukan untuk meluncurkan.

Amazon EC2 secara otomatis mengisi ulang snapshot yang Anda miliki saat menggunakannya untuk meluncurkan instans dari AMI yang mengaktifkan Peluncuran Cepat EC2.

Akun apa pun yang memiliki akses ke AMI dengan Peluncuran Cepat EC2 diaktifkan dapat memperoleh manfaat dari pengurangan waktu peluncuran. Ketika pemilik AMI memberikan akses bagi Anda untuk meluncurkan instans, snapshot yang telah disediakan sebelumnya diambil dari akun pemilik AMI.

Jika AMI yang mendukung Peluncuran Cepat EC2 dibagikan dengan Anda, Anda dapat mengaktifkan atau menonaktifkan peluncuran lebih cepat di AMI bersama sendiri. Jika Anda mengaktifkan AMI bersama untuk Peluncuran Cepat EC2, Amazon EC2 akan membuat snapshot yang telah disediakan

sebelumnya secara langsung di akun Anda. Jika Anda menghabiskan snapshot di akun Anda, Anda masih dapat menggunakan snapshot dari akun pemilik AMI.

> **Note**
>
> Peluncuran Cepat EC2 menghapus snapshot yang telah disediakan sebelumnya segera setelah dikonsumsi oleh peluncuran untuk meminimalkan biaya penyimpanan dan mencegah penggunaan kembali. Namun, jika snapshot yang dihapus cocok dengan aturan retensi, Keranjang Sampah secara otomatis mempertahankannya. Kami menyarankan Anda meninjau cakupan aturan retensi Keranjang Sampah Anda sehingga hal ini tidak terjadi. Untuk informasi selengkapnya, lihat Recycle Bin di Panduan Pengguna Amazon EBS.
> Fitur ini tidak sama dengan Pemulihan snapshot cepat EBS. Anda harus secara eksplisit mengaktifkan pemulihan snapshot cepat EBS dengan basis per snapshot, dan memiliki biaya tersendiri.

Video berikut menunjukkan cara mengonfigurasi AMI Windows Anda untuk peluncuran lebih cepat dengan ikhtisar singkat tentang istilah kunci terkait dan definisinya: Meluncurkan instans Windows EC2 hingga 65% lebih cepat aktif. AWS

Biaya sumber daya

Tidak ada biaya layanan untuk mengkonfigurasi AMI Windows untuk Peluncuran Cepat EC2. Namun, harga standar berlaku untuk AWS sumber daya dasar apa pun yang digunakan Amazon EC2. Untuk mempelajari lebih lanjut tentang biaya sumber daya terkait dan cara mengelolanya, lihat Kelola biaya untuk sumber daya dasar Peluncuran EC2 Cepat.

Daftar Isi


- Istilah kunci
- EC2 Prasyarat Peluncuran Cepat untuk Windows
- Konfigurasikan setelan Peluncuran Cepat EC2 untuk AMI Amazon EC2 Windows Server
- Lihat AMI dengan EC2 Fast Launch diaktifkan
- Kelola biaya untuk sumber daya dasar Peluncuran EC2 Cepat
- Pantau Peluncuran EC2 Cepat
- Service-linked peran untuk Peluncuran Cepat EC2
- Memecahkan Masalah Peluncuran Cepat Windows EC2

## Istilah kunci

Fitur Peluncuran Cepat EC2 menggunakan istilah kunci berikut:

Snapshot yang telah tersedia

Cuplikan instance yang diluncurkan dari Windows AMI dengan EC2 Fast Launch diaktifkan, dan yang telah menyelesaikan langkah-langkah peluncuran Windows berikut, me-reboot sesuai kebutuhan.

- Sysprep specialize
- Windows Out of Box Experience (OOBE)

Ketika langkah-langkah ini selesai, Peluncuran Cepat EC2 menghentikan instans, dan membuat snapshot yang nantinya digunakan untuk peluncuran lebih cepat dari AMI, berdasarkan konfigurasi Anda.

Frekuensi peluncuran

Mengontrol jumlah snapshot yang telah tersedia yang dapat diluncurkan Amazon EC2 dalam jangka waktu yang ditentukan. Saat Anda mengaktifkan Peluncuran Cepat EC2 untuk AMI, Amazon EC2 akan membuat kumpulan awal snapshot yang telah disediakan sebelumnya di latar belakang. Misalnya, jika frekuensi peluncuran diatur ke lima peluncuran per jam, yang merupakan default, maka Peluncuran Cepat EC2 membuat set awal lima snapshot yang telah disediakan sebelumnya.

Saat Amazon EC2 meluncurkan instans dari AMI dengan Peluncuran Cepat EC2 diaktifkan, Amazon EC2 menggunakan salah satu snapshot yang telah disediakan sebelumnya untuk mengurangi waktu peluncuran. Saat snapshot digunakan, mereka secara otomatis akan diisi ulang, hingga jumlah yang ditentukan oleh frekuensi peluncuran.

Jika Anda memperkirakan ada lonjakan jumlah instans yang diluncurkan dari AMI Anda – misalnya selama acara khusus, – Anda dapat meningkatkan frekuensi peluncuran terlebih dahulu untuk memperhitungkan instans tambahan yang akan Anda perlukan. Ketika tingkat peluncuran Anda kembali normal, Anda dapat menurunkan frekuensi kembali.

Ketika Anda mengalami jumlah peluncuran yang lebih tinggi daripada yang diperkirakan, Anda mungkin menggunakan semua snapshot yang telah tersedia yang ada. Hal ini tidak menyebabkan peluncuran menjadi gagal. Namun, dapat mengakibatkan beberapa instans mengalami proses peluncuran standar, sampai snapshot diisi ulang.

Jumlah sumber daya target

Jumlah snapshot yang telah disediakan sebelumnya untuk tetap tersedia untuk Amazon EC2 Windows Server AMI dengan Peluncuran Cepat EC2 diaktifkan.

Maksimal peluncuran paralel

Mengontrol berapa banyak instans Amazon EC2 yang dapat diluncurkan secara bersamaan untuk membuat snapshot yang telah disediakan sebelumnya untuk Peluncuran Cepat EC2. Jika jumlah sumber daya target Anda lebih tinggi dari peluncuran paralel maksimum yang telah Anda tetapkan, Amazon EC2 meluncurkan jumlah instans yang ditentukan oleh Maksimal peluncuran paralel untuk mulai membuat snapshot. Saat instans tersebut menyelesaikan proses, Amazon EC2 mengambil snapshot dan menghentikan instans. Lalu meluncurkan lebih banyak instans hingga jumlah total snapshot yang tersedia telah mencapai jumlah target sumber daya. Nilai untuk Maksimal peluncuran paralel harus 6 atau lebih besar.

## EC2 Prasyarat Peluncuran Cepat untuk Windows

Sebelum Anda mengatur Peluncuran EC2 Cepat, verifikasi bahwa Anda telah memenuhi prasyarat berikut yang diperlukan untuk membuat snapshot untuk di: AMIs Akun AWS

- Jika Anda menyediakan templat peluncuran kustom saat mengonfigurasi Peluncuran EC2 Cepat, layanan akan menggunakan VPC dan pengaturan konfigurasi lain yang telah Anda tetapkan dalam templat peluncuran. Untuk informasi selengkapnya, lihat Gunakan template peluncuran saat Anda mengatur EC2 Fast Launch.
- Jika tidak menggunakan templat peluncuran khusus untuk mengonfigurasi setelan, Anda harus melampirkan EC2FastLaunchFullAccess kebijakan tersebut ke peran IAM saat ini sebelum mengaktifkan Peluncuran EC2 Cepat. Kemudian layanan secara otomatis membuat CloudFormation tumpukan dengan sumber daya berikut di Anda Akun AWS.
  - Awan pribadi virtual (VPC)
  - Subnet pribadi di beberapa Availability Zone
  - Template peluncuran yang dikonfigurasi dengan Layanan Metadata Instance Versi 2 () IMDSv2
  - Grup keamanan tanpa aturan masuk atau keluar
- Peluncuran EC2 Cepat Pribadi AMIs harus mendukung eksekusi skrip data pengguna.
- Untuk mengonfigurasi Peluncuran EC2 Cepat untuk AMI, Anda harus membuat AMI menggunakan Sysprep opsi shutdown. Fitur EC2 Fast Launch saat ini tidak mendukung AMIs yang dibuat dari instance yang sedang berjalan.

Untuk membuat AMI menggunakan Sysprep, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.

- Untuk mengaktifkan Peluncuran EC2 Cepat untuk AMI terenkripsi yang menggunakan kunci terkelola pelanggan untuk enkripsi, Anda harus memberikan peran terkait layanan untuk izin Peluncuran EC2 Cepat untuk menggunakan CMK. Untuk informasi selengkapnya, lihat the section called “Akses ke kunci yang dikelola pelanggan”.
- Kuota default untuk peluncuran paralel Max di semua AMIs dalam an Akun AWS adalah 40 per Wilayah. Anda dapat meminta peningkatan Kuota Layanan untuk akun Anda, sebagai berikut.

  1. Buka konsol Service Quotas di https://console.aws.amazon.com/servicequotas/.
  2. Di panel navigasi, pilih Layanan AWS.
  3. Di bilah pencarian, masukkan **`EC2 Fast Launch`**, dan pilih hasilnya.
  4. Pilih link untuk peluncuran instance Parallel untuk membuka halaman detail kuota layanan.
  5. Pilih Permintaan peningkatan di tingkat akun.

  Untuk informasi selengkapnya, lihat Meminta peningkatan kuota di Panduan Pengguna Service Quotas.

## Konfigurasikan setelan Peluncuran Cepat EC2 untuk AMI Amazon EC2 Windows Server

Anda dapat mengonfigurasi EC2 Fast Launch untuk AMI Windows yang Anda miliki, atau AMI yang dibagikan dengan Anda dari Konsol Manajemen AWS, API, SDK CloudFormation, atau AWS Command Line Interface ()AWS CLI. Sebelum mengonfigurasi Peluncuran Cepat EC2, verifikasi bahwa AMI memenuhi semua prasyarat yang diperlukan untuk membuat snapshot yang telah disediakan sebelumnya. Untuk informasi selengkapnya, lihat EC2 Prasyarat Peluncuran Cepat untuk Windows.

Saat Anda mengaktifkan peluncuran yang lebih cepat untuk instans Windows, Amazon EC2 memeriksa untuk memastikan bahwa Anda memiliki izin yang diperlukan untuk meluncurkan instans dari AMI dan Template Peluncuran yang ditentukan (jika disediakan), termasuk izin untuk AMI terenkripsi. Untuk mencegah kesalahan selama proses peluncuran instans, layanan memvalidasi izin Anda sebelum Peluncuran Cepat EC2 diaktifkan. Jika Anda tidak memiliki izin yang diperlukan, layanan mengembalikan kesalahan, dan tidak mengaktifkan Peluncuran Cepat EC2.

EC2 Fast Launch terintegrasi dengan EC2 Image Builder untuk membantu Anda membuat gambar khusus dengan EC2 Fast Launch diaktifkan. Untuk informasi selengkapnya, lihat Membuat pengaturan distribusi untuk AMI Windows dengan Peluncuran Cepat EC2 yang diaktifkan (AWS CLI) di Panduan Pengguna EC2 Image Builder.

Aktifkan Peluncuran Cepat EC2

Sebelum mengubah pengaturan ini, pastikan AMI Anda, dan Wilayah yang Anda jalankan memenuhi semua EC2 Prasyarat Peluncuran Cepat untuk Windows.

Console

Untuk mengaktifkan Peluncuran Cepat EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, di bagian Gambar, pilih AMI.
3. Pilih AMI yang akan diperbarui dengan memilih kotak centang di samping Nama.
4. Dari menu Tindakan di atas daftar AMI, pilih Konfigurasikan peluncuran cepat. Ini membuka halaman Konfigurasi peluncuran cepat, tempat Anda mengonfigurasi pengaturan untuk EC2 Fast Launch.
5. Untuk mulai menggunakan snapshot yang telah disediakan sebelumnya untuk meluncurkan instans dari AMI Windows Anda lebih cepat, pilih kotak centang Aktifkan peluncuran cepat untuk Windows.
6. Dari daftar drop-down Tetapkan frekuensi peluncuran yang diantisipasi, pilih nilai untuk menentukan jumlah snapshot yang dibuat dan dipertahankan untuk mencukupi volume peluncuran instans yang Anda harapkan.
7. Setelah selesai membuat perubahan, pilih Simpan perubahan.

> **Note**
>
> Jika Anda perlu menggunakan template peluncuran untuk menentukan VPC, atau untuk mengonfigurasi pengaturan metadata untuk IMDSv2, lihat. Gunakan template peluncuran saat Anda mengatur EC2 Fast Launch

AWS CLI

Untuk mengaktifkan Peluncuran Cepat EC2

Gunakan perintah enable-fast-launch berikut untuk mengaktifkan Peluncuran Cepat EC2 untuk AMI yang ditentukan, meluncurkan enam instans paralel untuk pra-penyediaan.

```
aws ec2 enable-fast-launch \
    --image-id ami-0abcdef1234567890  \
    --max-parallel-launches 6 \
    --resource-type snapshot
```

Berikut ini adalah output contoh.

```
{
 "ImageId": "ami-0abcdef1234567890",
 "ResourceType": "snapshot",
 "SnapshotConfiguration": {
     "TargetResourceCount": 10
 },
 "LaunchTemplate": {},
 "MaxParallelLaunches": 6,
 "OwnerId": "0123456789123",
 "State": "enabling",
 "StateTransitionReason": "Client.UserInitiated",
 "StateTransitionTime": "2022-01-27T22:16:03.199000+00:00"
}
```

PowerShell

Untuk mengaktifkan Peluncuran Cepat EC2

Gunakan Enable-EC2FastLaunch cmdlet untuk mengaktifkan EC2 Fast Launch untuk AMI yang ditentukan, meluncurkan enam instans paralel untuk pra-penyediaan.

```
Enable-EC2FastLaunch `
 -ImageId ami-0abcdef1234567890 `
 -MaxParallelLaunch 6 `
 -Region us-west-2 `
 -ResourceType snapshot
```

Berikut ini adalah output contoh.

```
ImageId               : ami-0abcdef1234567890
 LaunchTemplate        :
 MaxParallelLaunches   : 6
 OwnerId               : 0123456789123
 ResourceType          : snapshot
 SnapshotConfiguration : Amazon.EC2.Model.FastLaunchSnapshotConfigurationResponse
 State                 : enabling
 StateTransitionReason : Client.UserInitiated
 StateTransitionTime   : 2/25/2022 12:24:11 PM
```

### Nonaktifkan Peluncuran Cepat EC2

Sebelum mengubah pengaturan ini, pastikan AMI Anda, dan Wilayah yang Anda jalankan memenuhi semua EC2 Prasyarat Peluncuran Cepat untuk Windows.

#### Console

Untuk menonaktifkan Peluncuran Cepat EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, di bagian Gambar, pilih AMI.
3. Pilih AMI yang akan diperbarui dengan memilih kotak centang di samping Nama.
4. Dari menu Tindakan di atas daftar AMI, pilih Konfigurasikan peluncuran cepat. Ini membuka halaman Konfigurasi peluncuran cepat, tempat Anda mengonfigurasi pengaturan untuk EC2 Fast Launch.
5. Kosongkan kotak centang Aktifkan peluncuran cepat untuk Windows untuk menonaktifkan Peluncuran Cepat EC2 dan untuk menghapus snapshot yang telah disediakan sebelumnya. Ini membuat AMI menggunakan proses peluncuran standar untuk setiap instans untuk seterusnya.

   > **Note**
   >
   > Saat Anda menonaktifkan optimisasi gambar Windows, snapshot yang telah tersedia akan dihapus secara otomatis. Langkah ini harus diselesaikan sebelum Anda dapat mulai menggunakan fitur ini lagi.

6. Setelah selesai membuat perubahan, pilih Simpan perubahan.

AWS CLI

Untuk menonaktifkan Peluncuran Cepat EC2

Gunakan perintah disable-fast-launch berikut untuk menonaktifkan Peluncuran Cepat EC2 pada AMI yang ditentukan, dan membersihkan snapshot yang sudah disediakan sebelumnya.

```
aws ec2 disable-fast-launch --image-id ami-01234567890abcedf
```

Berikut ini adalah output contoh.

```
{
    "ImageId": "ami-01234567890abcedf",
    "ResourceType": "snapshot",
    "SnapshotConfiguration": {},
    "LaunchTemplate": {
        "LaunchTemplateId": "lt-01234567890abcedf",
        "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
        "Version": "1"
    },
    "MaxParallelLaunches": 6,
    "OwnerId": "0123456789123",
    "State": "disabling",
    "StateTransitionReason": "Client.UserInitiated",
    "StateTransitionTime": "2022-01-27T22:47:29.265000+00:00"
}
```

PowerShell

Untuk menonaktifkan Peluncuran Cepat EC2

Gunakan Disable-EC2FastLaunch cmdlet untuk menonaktifkan Peluncuran Cepat EC2 pada AMI yang ditentukan, dan bersihkan snapshot yang sudah disediakan sebelumnya.

```
Disable-EC2FastLaunch -ImageId ami-0abcdef1234567890
```

Berikut ini adalah output contoh.

```
ImageId               : ami-0abcdef1234567890
 LaunchTemplate        :
 Amazon.EC2.Model.FastLaunchLaunchTemplateSpecificationResponse
 MaxParallelLaunches   : 6
```

```
OwnerId               : 0123456789123
ResourceType          : snapshot
SnapshotConfiguration :
State                 : disabling
StateTransitionReason : Client.UserInitiated
StateTransitionTime   : 2/25/2022 1:10:08 PM
```

Gunakan template peluncuran saat Anda mengatur EC2 Fast Launch

Dengan templat peluncuran, Anda dapat mengonfigurasi serangkaian parameter peluncuran yang digunakan Amazon EC2 setiap kali meluncurkan instans dari templat tersebut. Anda dapat menentukan hal-hal seperti AMI yang akan digunakan untuk gambar dasar, tipe instans, penyimpanan, pengaturan jaringan, dan lainnya.

Templat peluncuran bersifat opsional, kecuali untuk kasus spesifik berikut, di mana Anda harus menggunakan templat peluncuran untuk AMI Windows Anda saat Anda mengonfigurasi peluncuran yang lebih cepat:

- Anda harus menggunakan template peluncuran untuk menentukan VPC yang ada untuk AMI Windows Anda. Ini tidak berlaku jika Anda menggunakan VPC default untuk Anda Akun AWS.
- Jika akun Anda memiliki kebijakan yang memberlakukan IMDSv2 untuk instans Amazon EC2, Anda harus membuat templat peluncuran yang menentukan konfigurasi metadata untuk menerapkan ImDSv2.

  Gunakan template peluncuran yang menyertakan konfigurasi metadata Anda dari konsol EC2, atau saat Anda menjalankan perintah enable-fast-launch di AWS CLI, atau memanggil tindakan API. EnableFastLaunch

EC2 Fast Launch tidak mendukung konfigurasi berikut saat Anda menggunakan template peluncuran. Jika Anda menggunakan template peluncuran untuk EC2 Fast Launch, Anda tidak boleh menentukan salah satu dari berikut ini:

- Skrip data pengguna
- Perlindungan pengakhiran
- Metadata dinonaktifkan
- Opsi spot
- Perilaku shutdown yang mengakhiri instance

- Tag sumber daya untuk antarmuka jaringan, grafik elastis, atau permintaan instans spot

Tentukan VPC

Langkah 1: Buat templat peluncuran

Buat templat peluncuran yang menetapkan detail berikut untuk instans Windows Anda:

- Subnet VPC.
- Tipe instans `t3.xlarge`.

Untuk informasi selengkapnya, lihat Buat template peluncuran Amazon EC2.

Langkah 2: Tentukan template peluncuran untuk AMI Peluncuran Cepat EC2 Anda

Console

Untuk menentukan template peluncuran untuk EC2 Fast Launch

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, di bagian Gambar, pilih AMI.
3. Pilih AMI yang akan diperbarui dengan memilih kotak centang di samping Nama.
4. Dari menu Tindakan di atas daftar AMI, pilih Konfigurasikan peluncuran cepat. Ini membuka halaman Konfigurasi peluncuran cepat, tempat Anda mengonfigurasi pengaturan untuk EC2 Fast Launch.
5. Kotak Templat peluncuran melakukan penelusuran tersaring yang mencari templat peluncuran di akun Anda di Wilayah saat ini yang cocok dengan teks yang Anda masukkan. Tentukan semua atau sebagian nama atau ID templat peluncuran di kotak untuk menampilkan daftar templat peluncuran yang cocok. Misalnya, jika Anda memasukkan `fast` dalam kotak, Amazon EC2 mencari semua templat peluncuran di akun Anda di Wilayah saat ini yang memiliki nama “cepat”.

   Untuk membuat templat peluncuran baru, Anda dapat memilih Buat templat peluncuran.
6. Saat Anda memilih sebuah templat peluncuran, Amazon EC2 menampilkan versi default untuk templat tersebut di kotak Versi templat asal. Untuk menentukan versi yang berbeda, sorot versi default untuk menggantinya, dan masukkan nomor versi yang Anda inginkan di kotak.

7. Setelah selesai membuat perubahan, pilih Simpan perubahan.

AWS CLI

Untuk menentukan template peluncuran untuk EC2 Fast Launch

Gunakan perintah enable-fast-launch dengan `--launch-template` opsi, tentukan nama atau ID template peluncuran.

```
--launch-template LaunchTemplateName=my-launch-template
```

PowerShell

Untuk menentukan template peluncuran untuk EC2 Fast Launch

Gunakan Enable-EC2FastLaunch cmdlet dengan parameter - `LaunchTemplate_LaunchTemplateId` atau-`LaunchTemplate_LaunchTemplateName`.

```
-LaunchTemplate_LaunchTemplateName my-launch-template
```

Untuk informasi selengkapnya tentang templat peluncuran EC2, lihat Simpan parameter peluncuran instans di templat peluncuran Amazon EC2.

Pemeriksaan izin untuk Peluncuran Cepat EC2

Saat Anda mengaktifkan EC2 Fast Launch dengan template peluncuran, Anda dapat menentukan versi template bernomor. Anda juga dapat menggunakan `$Default` versi `$Latest` atau. Jika Anda menggunakan salah satu alias, alias dapat mengganti `iam:PassRole` izin `ec2:RunInstances` dan yang ingin Anda batasi.

Saat Anda mengaktifkan Peluncuran Cepat EC2, Amazon EC2 memeriksa izin Anda terhadap versi template peluncuran saat ini. `$Latest`dan `$Default` masing-masing memutuskan ke versi tertentu pada saat itu. Pemeriksaan ini berjalan sebelum Amazon EC2 mengaktifkan fitur tersebut. Ini memvalidasi izin yang diperlukan untuk meluncurkan instance, seperti `iam:PassRole` izin `ec2:RunInstances` dan. Untuk melakukan ini, EC2 Fast Launch mengeluarkan panggilan `RunInstances` dry run yang memeriksa izin Anda tanpa meluncurkan instance. Jika izin Anda tidak mengizinkan tindakan yang diperlukan, Amazon EC2 gagal dalam permintaan dan mengembalikan kesalahan yang menjelaskan izin yang hilang. Amazon EC2 tidak mengaktifkan Peluncuran Cepat EC2.

Setelah pemeriksaan awal ini selesai, EC2 Fast Launch membuat snapshot yang telah disediakan sebelumnya di latar belakang. Kemudian meluncurkan instance dengan menggunakan izin dari peran yang ditautkan layanan. Setiap kali meluncurkan instance, ia menggunakan versi yang `$Latest` atau `$Default` diselesaikan pada saat itu. Ini benar bahkan jika versi telah berubah setelah pemeriksaan awal. Itu tidak memeriksa ulang izin Anda. Akibatnya, seseorang yang dapat memperbarui template peluncuran dapat meneruskan peran IAM atau profil instance ke instance. Ini bisa terjadi bahkan jika mereka tidak memiliki `iam:PassRole` izin untuk peran itu.

Untuk memastikan bahwa Amazon EC2 selalu menggunakan versi yang divalidasi saat Anda mengaktifkan fitur, tentukan versi template peluncuran bernomor. Jangan gunakan `$Latest` atau`$Default`.

Jika Anda mengonfigurasi Peluncuran Cepat EC2 untuk menggunakan `$Latest` atau `$Default` meluncurkan versi template, sebaiknya batasi siapa yang dapat membuat dan mengelola versi template peluncuran. Gunakan kebijakan IAM untuk membatasi akses ke tindakan seperti `ec2:CreateLaunchTemplateVersion` dan`ec2:ModifyLaunchTemplate`.

## Lihat AMI dengan EC2 Fast Launch diaktifkan

Anda dapat menggunakan perintah describe-fast-launch-images di AWS CLI, atau Get-EC2FastLaunchImage Tools for PowerShell Cmdlet untuk mendapatkan detail AMI yang mengaktifkan EC2 Fast Launch.

Amazon EC2 memberikan detail berikut untuk setiap AMI Windows yang muncul dalam hasil:

- ID gambar untuk AMI dengan EC2 Fast Launch diaktifkan.
- Tipe sumber daya yang digunakan untuk pra-penyediaan AMI Windows terkait. Nilai yang didukung: `snapshot`.
- Konfigurasi snapshot, yang merupakan sekelompok parameter yang mengonfigurasi pra-penyediaan untuk AMI Windows terkait menggunakan snapshot.
- Meluncurkan informasi templat, termasuk ID, nama, dan versi templat peluncuran yang digunakan AMI terkait saat meluncurkan instans Window dari snapshot yang telah tersedia.
- Jumlah maksimum instans yang dapat diluncurkan pada saat yang sama untuk membuat sumber daya.
- ID pemilik untuk AMI terkait. Ini tidak diisi untuk AMI yang dibagikan dengan Anda.
- Status EC2 Fast Launch saat ini untuk AMI terkait. Nilai yang didukung meliputi: `enabling | enabling-failed | enabled | enabled-failed | disabling | disabling-failed`.

> **Note**
>
> Anda juga dapat melihat status saat ini yang ditampilkan di halaman Mengelola optimisasi gambar di konsol EC2, sebagai status optimisasi gambar.

- Alasan EC2 Fast Launch untuk AMI terkait berubah ke status saat ini.
- Waktu Peluncuran Cepat EC2 untuk AMI terkait berubah ke status saat ini.

AWS CLI

Untuk menemukan AMI yang dikonfigurasi untuk EC2 Fast Launch

Gunakan perintah describe-fast-launch-images berikut untuk menjelaskan detail untuk setiap AMI di akun yang dikonfigurasi untuk EC2 Fast Launch. Dalam contoh ini, hanya satu AMI di akun yang dikonfigurasi untuk EC2 Fast Launch.

```
aws ec2 describe-fast-launch-images
```

Berikut ini adalah output contoh.

```
{
    "FastLaunchImages": [
        {
            "ImageId": "ami-01234567890abcedf",
            "ResourceType": "snapshot",
            "SnapshotConfiguration": {},
            "LaunchTemplate": {
                "LaunchTemplateId": "lt-01234567890abcedf",
                "LaunchTemplateName": "EC2FastLaunchDefaultResourceCreation-
a8c6215d-94e6-441b-9272-dbd1f87b07e2",
                "Version": "1"
            },
            "MaxParallelLaunches": 6,
            "OwnerId": "0123456789123",
            "State": "enabled",
            "StateTransitionReason": "Client.UserInitiated",
            "StateTransitionTime": "2022-01-27T22:20:06.552000+00:00"
        }
    ]
```

```
}
```

PowerShell

Untuk menemukan AMI yang dikonfigurasi untuk EC2 Fast Launch

Gunakan Get-EC2FastLaunchImage cmdlet berikut untuk menjelaskan detail untuk setiap AMI di akun yang dikonfigurasi untuk Peluncuran Cepat EC2. Dalam contoh ini, hanya satu AMI di akun yang dikonfigurasi untuk EC2 Fast Launch.

```
Get-EC2FastLaunchImage -ImageId ami-0abcdef1234567890
```

Berikut ini adalah output contoh.

```
ImageId               : ami-0abcdef1234567890
LaunchTemplate        :
 Amazon.EC2.Model.FastLaunchLaunchTemplateSpecificationResponse
MaxParallelLaunches   : 6
OwnerId               : 012345678912
ResourceType          : snapshot
SnapshotConfiguration :
State                 : enabled
StateTransitionReason : Client.UserInitiated
StateTransitionTime   : 2/25/2022 12:54:43 PM
```

## Kelola biaya untuk sumber daya dasar Peluncuran EC2 Cepat

Tidak ada biaya layanan untuk mengkonfigurasi Windows AMIs untuk Peluncuran EC2 Cepat. Namun, saat Anda mengaktifkan Peluncuran EC2 Cepat untuk Amazon EC2 Windows AMI, harga standar berlaku untuk AWS sumber daya dasar yang EC2 digunakan Amazon untuk menyiapkan dan menyimpan snapshot yang telah disediakan sebelumnya. Anda dapat mengonfigurasi tag alokasi biaya untuk membantu Anda melacak dan mengelola biaya yang terkait dengan sumber daya Peluncuran EC2 Cepat. Untuk informasi selengkapnya tentang cara mengonfigurasi tag alokasi biaya, lihat Lacak biaya Peluncuran EC2 Cepat pada tagihan Anda.

Contoh berikut menunjukkan bagaimana biaya yang terkait dengan biaya snapshot Peluncuran EC2 Cepat mungkin dialokasikan.

Contoh skenario: Perusahaan AtoZ memiliki AMI Windows dengan volume root 50 GiB EBS. Mereka mengaktifkan Peluncuran EC2 Cepat untuk AMI mereka, dan menetapkan jumlah sumber daya target

menjadi lima. Selama sebulan, menggunakan Peluncuran EC2 Cepat untuk AMI mereka harganya sekitar $5,00, dan rincian biaya adalah sebagai berikut:

1. Saat AtoZ Example mengaktifkan Peluncuran EC2 Cepat, Amazon EC2 meluncurkan lima instans kecil. Setiap instans berjalan melalui langkah-langkah peluncuran Windows Sysprep dan OOBE, dan boot ulang sesuai kebutuhan. Proses ini memakan waktu beberapa menit untuk setiap instans (waktu dapat bervariasi, berdasarkan seberapa sibuk Wilayah atau Zona Ketersediaan (AZ), dan pada ukuran AMI).

   Biaya

   - Biaya runtime instans (atau runtime minimum, jika ada): lima instans
   - Biaya volume: lima volume root EBS
2. Saat proses pra-penyediaan selesai, EC2 Amazon mengambil snapshot instance, yang disimpan di Amazon S3. Snapshot biasanya disimpan selama 4–8 jam sebelum dikonsumsi oleh peluncuran. Dalam hal ini, biayanya kira-kira 0,02 USD hingga 0,05 USD per snapshot.

   Biaya

   - Penyimpanan snapshot (Amazon S3): lima snapshot
3. Setelah Amazon EC2 mengambil snapshot, itu menghentikan instance. Pada saat itu, instans tidak lagi dikenai biaya. Namun biaya volume EBS terus bertambah.

   Biaya

   - Volume EBS: biaya berlanjut untuk volume root EBS terkait.

> **Note**
>
> Biaya yang ditampilkan di sini hanya contoh. Biaya yang Anda bayar akan bervariasi, tergantung pada konfigurasi AMI dan paket harga Anda.

### Lacak biaya Peluncuran EC2 Cepat pada tagihan Anda

Tag alokasi biaya dapat membantu Anda mengatur AWS tagihan untuk mencerminkan biaya yang terkait dengan Peluncuran EC2 Cepat. Anda dapat menggunakan tag berikut yang EC2 ditambahkan Amazon ke sumber daya yang dibuatnya saat menyiapkan dan menyimpan snapshot yang telah disediakan sebelumnya untuk Peluncuran Cepat: EC2

Kunci tag: `CreatedBy`, Nilai: `EC2 Fast Launch`

Setelah Anda mengaktivasi tag tersebut di konsol Manajemen Penagihan dan Biaya, lalu mengatur laporan penagihan terperinci, kolom `user:CreatedBy` tersebut muncul di laporan. Kolom mencakup nilai dari semua layanan. Namun, jika Anda mengunduh file CSV, Anda dapat mengimpor data ke dalam spreadsheet, dan memfilter `EC2 Fast Launch` pada nilainya. Informasi ini juga muncul di AWS Cost and Usage Report saat tag diaktifkan.

Langkah 1: Aktivasi tag alokasi biaya yang ditentukan pengguna

Untuk menyertakan tag sumber daya di laporan biaya Anda, Anda harus mengaktivasi tag tersebut terlebih dahulu di konsol Manajemen Penagihan dan Biaya. Untuk informasi selengkapnya, lihat Mengaktifkan Tag Alokasi Biaya Buatan Pengguna dalam Panduan Pengguna AWS Manajemen Penagihan dan Biaya .

> **Note**
>
> Aktivasi dapat memakan waktu hingga 24 jam.

Langkah 2: Mengatur laporan biaya

Jika Anda sudah mengatur laporan biaya, kolom untuk tag Anda akan muncul saat laporan berikutnya setelah aktivasi selesai. Untuk mengatur laporan biaya untuk pertama kalinya, pilih salah satu dari hal berikut ini.

- Lihat Mengatur laporan alokasi biaya bulanan di Panduan Pengguna AWS Manajemen Penagihan dan Biaya .
- Lihat Membuat Laporan Biaya dan Penggunaan di Panduan Pengguna AWS Cost and Usage Report .

> **Note**
>
> Diperlukan waktu hingga 24 jam untuk AWS mulai mengirimkan laporan ke bucket S3 Anda.

Anda dapat mengonfigurasi Peluncuran EC2 Cepat untuk Windows AMIs AMIs yang Anda miliki, atau yang dibagikan dengan Anda dari EC2 konsol Amazon, API SDKs, CloudFormation,, atau

ec2 perintah di AWS CLI. Bagian berikut mencakup langkah-langkah konfigurasi untuk EC2 konsol Amazon dan AWS CLI.

Anda juga dapat membuat Windows kustom AMIs yang dikonfigurasi untuk EC2 Fast Launch dengan EC2 Image Builder. Untuk informasi selengkapnya, lihat Membuat setelan distribusi untuk AMI Windows dengan Peluncuran EC2 Cepat diaktifkan (AWS CLI).

## Pantau Peluncuran EC2 Cepat

Bagian ini mencakup cara memantau Amazon EC2 Windows Server AMIs di akun Anda yang mengaktifkan Peluncuran EC2 Cepat.

Pantau perubahan status Peluncuran EC2 Cepat dengan EventBridge

Saat status berubah untuk AMI Windows dengan Peluncuran EC2 Cepat diaktifkan, Amazon EC2 menghasilkan `EC2 Fast Launch State-change Notification` acara. Kemudian Amazon EC2 mengirimkan peristiwa perubahan status ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon CloudWatch Events).

Anda dapat membuat EventBridge aturan yang memicu satu atau beberapa tindakan sebagai respons terhadap peristiwa perubahan status. Misalnya, Anda dapat membuat EventBridge aturan yang mendeteksi kapan Peluncuran EC2 Cepat diaktifkan dan melakukan tindakan berikut:

- Mengirim pesan ke topik Amazon SNS yang memberi tahu para pelanggan.
- Menginvokasi fungsi Lambda yang melakukan beberapa tindakan.
- Mengirim data perubahan status ke Amazon Data Firehose untuk analitik.

Untuk informasi selengkapnya, lihat Membuat EventBridge aturan Amazon yang bereaksi terhadap peristiwa di Panduan EventBridge Pengguna Amazon.

Peristiwa perubahan status

Fitur Peluncuran EC2 Cepat memancarkan peristiwa perubahan status berformat JSON dengan upaya terbaik. Amazon EC2 mengirimkan acara ke EventBridge dalam waktu dekat. Bagian ini menjelaskan bidang peristiwa dan menunjukkan contoh format peristiwa.

**`EC2 Fast Launch State-change Notification`**

imageId

Mengidentifikasi AMI dengan perubahan status Peluncuran EC2 Cepat.

resourceType

Tipe sumber daya yang digunakan untuk pra-penyediaan. Nilai yang didukung: `snapshot`. Nilai default-nya adalah `snapshot`.

status

Status fitur EC2 Fast Launch saat ini untuk AMI yang ditentukan. Nilai-nilai yang valid meliputi:

- mengaktifkan - Anda telah mengaktifkan fitur Peluncuran EC2 Cepat untuk AMI, dan Amazon EC2 telah mulai membuat snapshot untuk proses pra-penyediaan.
- enabling-failed - Ada yang tidak beres yang menyebabkan proses pra-penyediaan gagal saat pertama kali Anda mengaktifkan Peluncuran Cepat EC2 untuk AMI. Hal ini dapat terjadi kapan saja selama proses pra-penyediaan.
- diaktifkan - Fitur Peluncuran EC2 Cepat diaktifkan. Status berubah menjadi `enabled` segera setelah Amazon EC2 membuat snapshot pra-penyediaan pertama untuk AMI Peluncuran Cepat EC2 yang baru diaktifkan. Jika AMI sudah diaktifkan dan melalui pra-penyediaan lagi, perubahan status akan segera terjadi.
- enabled-failed - Status ini hanya berlaku jika ini bukan pertama kalinya AMI EC2 Peluncuran Cepat Anda melewati proses pra-penyediaan. Hal ini dapat terjadi jika fitur EC2 Fast Launch dinonaktifkan dan kemudian diaktifkan lagi, atau jika ada perubahan konfigurasi atau kesalahan lain setelah pra-penyediaan selesai untuk pertama kalinya.
- menonaktifkan - Pemilik AMI telah mematikan fitur Peluncuran EC2 Cepat untuk AMI, dan Amazon EC2 telah memulai proses pembersihan.
- dinonaktifkan - Fitur Peluncuran EC2 Cepat dinonaktifkan. Status berubah menjadi `disabled` segera setelah Amazon EC2 menyelesaikan proses pembersihan.
- gagal-menonaktifkan – Terjadi kesalahan yang menyebabkan proses pembersihan gagal. Ini berarti bahwa beberapa snapshot pra-penyediaan mungkin masih ada di akun.

stateTransitionReason

Alasan bahwa negara berubah untuk EC2 Fast Launch AMI.

> **Note**
>
> Semua bidang dalam pesan peristiwa ini diperlukan.

Contoh berikut menunjukkan AMI Peluncuran EC2 Cepat yang baru diaktifkan yang telah meluncurkan instance pertama untuk memulai proses pra-penyediaan. Pada saat ini, statusnya adalah `enabling`. Setelah Amazon EC2 membuat snapshot pra-penyediaan pertama, status berubah menjadi. `enabled`

```
{
 "version": "0",
 "id": "01234567-0123-0123-0123-012345678901",
 "detail-type": "EC2 Fast Launch State-change Notification",
 "source": "aws.ec2",
 "account": "123456789012",
 "time": "2022-08-31T20:30:12Z",
 "region": "us-east-1",
 "resources": [
  "arn:aws:ec2:us-east-1:123456789012:image/ami-123456789012"
 ],
 "detail": {
  "imageId": "ami-123456789012",
  "resourceType": "snapshot",
  "state": "enabling",
  "stateTransitionReason": "Client.UserInitiated"
 }
}
```

Pantau metrik Peluncuran EC2 Cepat dengan CloudWatch

Amazon EC2 AMIs dengan Peluncuran EC2 Cepat mengaktifkan kirim metrik ke Amazon CloudWatch. Anda dapat menggunakan, API Konsol Manajemen AWS AWS CLI, atau API untuk mencantumkan metrik yang dikirimkan EC2 Fast Launch. CloudWatch `AWS/EC2`Namespace mencakup metrik Peluncuran EC2 Cepat berikut:

| Metrik | Deskripsi |
| --- | --- |
| NumberOfAvailableFastLaunchSnapshots | Jumlah snapshot pra-penyediaan yang tersedia per EC2 Peluncuran Cepat mengaktifkan AMI. |
| NumberOfInstancesFastLaunched | Jumlah instans per Peluncuran EC2 Cepat memungkinkan AMI yang diluncurkan dari snapshot yang telah disediakan sebelumnya. |

| Metrik | Deskripsi |
| --- | --- |
| NumberOfInstancesNotFastLaunched | Jumlah instans per Peluncuran EC2 Cepat memungkinkan AMI yang menghasilkan boot dingin karena kurangnya snapshot pra-penye diaan yang tersedia pada waktu peluncuran. |
| FastLaunchSnapshotUsedToRefillStartTime | Stempel waktu saat Amazon EC2 meluncurk an gambar baru dari Peluncuran EC2 Cepat memungkinkan AMI membuat snapshot lain setelah snapshot yang ada digunakan. |
| FastLaunchSnapshotCreationTime | Mengukur waktu yang dibutuhkan Amazon EC2 untuk meluncurkan instance dan membuat snapshot untuk AMI yang mengaktifkan Peluncuran EC2 Cepat. |

## Service-linked peran untuk Peluncuran Cepat EC2

Amazon EC2 menggunakan peran tertaut layanan untuk izin yang diperlukan untuk memanggil Layanan AWS lain mewakili Anda. Peran terkait layanan adalah jenis peran IAM unik yang ditautkan langsung ke. Layanan AWS Service-linked peran menyediakan cara aman untuk mendelegasikan izin Layanan AWS karena hanya layanan tertaut yang dapat mengambil peran terkait layanan. Untuk informasi selengkapnya tentang cara Amazon EC2 menggunakan peran IAM, termasuk peran tertaut layanan, lihat IAM role untuk Amazon EC2.

Amazon EC2 menggunakan peran tertaut layanan bernama AWSServiceRoleForEC2FastLaunch untuk membuat dan mengelola sepaket snapshot pra-penyediaan yang mengurangi waktu yang diperlukan untuk meluncurkan instans dari AMI Windows Anda.

Izin yang diberikan oleh AWSServiceRoleForEC2FastLaunch

Peran terkait layanan AWSServiceRoleForEC2FastLaunch memercayai layanan berikut untuk mengambil peran tersebut:

- `ec2fastlaunch.amazonaws.com`

Amazon EC2 menggunakan kebijakan yang dikelola EC2FastLaunchServiceRolePolicy untuk menyelesaikan tindakan berikut:

- AWS CloudFormation— Izinkan Peluncuran Cepat EC2 untuk mendapatkan deskripsi tumpukan terkait CloudFormation .
- Amazon CloudWatch — Posting data metrik yang terkait dengan Peluncuran Cepat EC2 ke namespace Amazon EC2.
- Amazon EC2 — Akses diberikan untuk Peluncuran Cepat EC2 untuk melakukan tindakan berikut:
  - Luncurkan instans dari AMI Amazon EC2 Windows Server dengan EC2 Fast Launch diaktifkan, untuk melakukan langkah-langkah penyediaan. Selain itu, tentukan pola sumber daya yang memungkinkan `ec2:RunInstances` AMI yang terkait dengan Manajer Lisensi.
  - Hentikan dan hentikan instance yang diluncurkan oleh EC2 Fast Launch setelah membuat snapshot yang telah disediakan sebelumnya.
  - Jelaskan sumber daya tipe gambar dan instans yang digunakan untuk meluncurkan instans dari AMI Amazon EC2 Windows Server dengan EC2 Fast Launch diaktifkan dan buat snapshot darinya.
  - Jelaskan sumber daya template peluncuran dan peluncuran instance dari template peluncuran.
  - Jelaskan instance, atribut instance dan status instance, volume dan atribut volume.
  - Jelaskan antarmuka jaringan.
  - Hapus sumber daya yang dibuat EC2 Fast Launch, termasuk snapshot, templat peluncuran, volume, dan antarmuka jaringan.
  - Tandai sumber daya yang dibuat EC2 Fast Launch untuk meluncurkan dan pra-penyediaan, instans Windows, dan membuat snapshot untuk digunakan proses peluncuran akhir.
- Amazon EventBridge — Termasuk akses untuk membuat aturan EventBridge acara dan mengambil detail tentang atau menghapus aturan yang dibuat. EC2 Fast Launch juga dapat mendapatkan daftar layanan target yang menerima acara Peluncuran Cepat EC2 yang diteruskan berdasarkan aturan acara, dan menambahkan layanan target ke atau menghapusnya dari aturan acara yang dibuat.
- IAM - Memungkinkan EC2 Fast Launch untuk membuat peran `EC2FastLaunchServiceRolePolicy` terkait layanan, untuk mendapatkan dan menggunakan profil instance yang berisi namanya`ec2fastlaunch`, dan untuk meluncurkan instans atas nama Anda menggunakan profil instance dari template peluncuran Anda.
- AWS KMS— Termasuk akses untuk membuat hibah dan daftar hibah yang dibuat oleh EC2 Fast Launch yang dapat dihentikan. Juga untuk menjelaskan atau menggunakan kunci untuk tujuan

mengenkripsi atau mendekripsi volume yang dilampirkan ke instans yang dibuat EC2 Fast Launch, dan untuk menghasilkan kunci data yang bukan teks biasa.

Untuk melihat izin kebijakan ini, lihat EC2FastLaunchServiceRolePolicy di Referensi Kebijakan AWS Terkelola.

Untuk informasi selengkapnya tentang penggunaan kebijakan terkelola untuk Amazon EC2, lihat AWS kebijakan terkelola untuk Amazon EC2.

Buat peran tertaut layanan

Anda tidak perlu membuat peran tertaut layanan secara manual. Saat Anda mulai menggunakan EC2 Fast Launch untuk AMI Anda, Amazon EC2 membuat peran terkait layanan untuk Anda, jika itu belum ada.

Jika peran terkait layanan dihapus dari akun Anda, Anda dapat mengaktifkan EC2 Fast Launch untuk AMI Windows lain untuk membuat ulang peran di akun Anda. Atau, Anda dapat menonaktifkan Peluncuran Cepat EC2 untuk AMI Anda saat ini, lalu mengaktifkannya lagi. Namun, menonaktifkan fitur ini akan membuat AMI Anda menggunakan proses peluncuran standar untuk semua instans baru sementara Amazon EC2 menghapus semua snapshot pra-penyediaan Anda. Setelah semua snapshot yang telah disediakan sebelumnya hilang, Anda dapat mengaktifkan menggunakan EC2 Fast Launch untuk AMI Anda lagi.

Akses ke kunci yang dikelola pelanggan

Untuk mengaktifkan Peluncuran Cepat EC2 untuk AMI terenkripsi yang menggunakan kunci yang dikelola pelanggan untuk enkripsi, Anda harus memberikan izin AWSServiceRoleForEC2FastLaunch peran untuk menggunakan CMK. Untuk melakukan ini, panggil perintah create-grant. Untuk `--grantee-principal`, tentukan ARN untuk AWSServiceRoleForEC2FastLaunch peran di akun Anda. Untuk `--operations`, tentukan `CreateGrant`.

```
aws kms create-grant \
    --key-id arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/AWSServiceRoleForEC2FastLaunch \
    --operations CreateGrant
```

Edit peran tertaut layanan

Amazon EC2 tidak mengizinkan Anda untuk mengubah peran tertaut layanan AWSServiceRoleForEC2FastLaunch. Setelah Anda membuat peran terkait layanan, Anda tidak dapat mengubah nama peran, karena berbagai entitas mungkin mereferensikan peran tersebut. Namun, Anda dapat mengubah deskripsi peran dengan menggunakan IAM. Untuk informasi selengkapnya, lihat Mengedit peran terkait layanan dalam Panduan Pengguna IAM.

Hapus peran tertaut layanan

Anda dapat menghapus peran tertaut layanan hanya setelah menghapus semua sumber daya terkait terlebih dahulu. Ini melindungi sumber daya Amazon EC2 yang terkait dengan AMI Amazon EC2 Windows Server Anda dengan EC2 Fast Launch diaktifkan, karena Anda tidak dapat menghapus izin untuk mengakses sumber daya secara tidak sengaja.

Gunakan konsol IAM AWS CLI, atau AWS API untuk menghapus peran terkait AWSServiceRoleForEC2FastLaunch layanan. Untuk informasi selengkapnya, lihat Mengh apus peran terkait layanan di Panduan Pengguna IAM.

Wilayah yang Didukung

Amazon EC2 mendukung peran terkait layanan Peluncuran Cepat EC2 di semua Wilayah tempat layanan Amazon EC2 tersedia.

## Memecahkan Masalah Peluncuran Cepat Windows EC2

Skenario pemecahan masalah

Skenario berikut dapat membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat mencoba mengaktifkan Peluncuran Cepat EC2.

Tidak dapat menghentikan instance untuk pembuatan snapshot

Deskripsi

Saat Anda mengaktifkan Peluncuran Cepat EC2, layanan akan meluncurkan serangkaian instans yang digunakan untuk membuat snapshot yang telah disediakan sebelumnya. Setiap contoh diberikan 30 menit untuk menyelesaikan proses. Jika salah satu instans berhasil diselesaikan, maka layanan menetapkan status Peluncuran Cepat untuk AMI ke`Enabled`. Namun, jika instance gagal menyelesaikan proses dalam waktu yang ditentukan, dan tidak ada instans lain yang menyelesaikan

proses, layanan menghentikan semua instans dan menetapkan status Peluncuran Cepat untuk AMI `enabling_failed` dan alasan status Peluncuran Cepat sebagai berikut:

```
Unable to stop instance ID=i-1234567890abcdef0 for snapshot creation.
```

Penyebab

Paling sering, ini disebabkan oleh mencoba mengaktifkan EC2 Fast Launch untuk AMI Windows yang dibuat dari instance yang sedang berjalan, atau AMI yang tidak memenuhi semua. EC2 Prasyarat Peluncuran Cepat

Solusi

Pastikan AMI yang Anda gunakan memenuhi semuaEC2 Prasyarat Peluncuran Cepat.

Untuk mengkonfigurasi EC2 Fast Launch untuk AMI, Anda harus membuat AMI menggunakan Sysprep dengan opsi shutdown. Untuk informasi selengkapnya, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.

Anda telah mencapai batas VPC

Deskripsi

Jika Anda tidak menggunakan template peluncuran untuk menentukan VPC yang ada, dan tidak memiliki VPC default yang ditentukan untuk akun Anda, layanan akan secara otomatis membuat CloudFormation tumpukan yang menyertakan VPC dan sumber daya lainnya, seperti yang dijelaskan dalam. EC2 Prasyarat Peluncuran Cepat

Penyebab

Anda telah mencapai jumlah maksimum VPC yang diizinkan di Wilayah Anda Akun AWS , dan Anda belum menentukan VPC yang ada untuk digunakan Peluncuran Cepat EC2. Hal ini menyebabkan proses gagal.

Solusi

Anda dapat mengatasi masalah ini dengan salah satu opsi berikut:

- Anda dapat meminta kenaikan kuota
- Anda dapat menyediakan template peluncuran yang menentukan VPC yang ada

Untuk meminta peningkatan jumlah VPC yang dapat ditentukan akun Anda per Wilayah, ikuti langkah-langkah berikut:

1. Buka konsol Service Quotas di https://console.aws.amazon.com/servicequotas/.
2. Di Dasbor Konsol Layanan, pilih Amazon Virtual Private Cloud (Amazon VPC). Ini membuka kuota layanan VPC.
3. Saring `VPCs per Region` untuk langsung masuk ke kuota.
4. Pilih VPC per Wilayah, dan pilih Min ta peningkatan di tingkat akun.

Jika Anda memiliki permintaan kuota yang mendesak, atau jika permintaan kenaikan kuota Anda ditolak, hubungi Dukungan untuk mendapatkan bantuan. Untuk informasi selengkapnya, lihat Meminta peningkatan kuota di Panduan Pengguna Service Quotas.

Izin tidak mencukupi untuk mengaktifkan Peluncuran Cepat EC2

Deskripsi

Saat Anda mengaktifkan Peluncuran Cepat EC2 untuk pertama kalinya tanpa menentukan Template Peluncuran, EC2 Fast Launch membuat CloudFormation tumpukan milik layanan dengan sumber daya default layanan. Namun, CloudFormation template akan gagal diterapkan jika prinsipal IAM Anda (peran atau pengguna) tidak memiliki izin yang diperlukan.

Pesan log mungkin terlihat seperti berikut:

```
Can't enable EC2 Fast Launch. The IAM credentials that you are using do not have
 sufficient permissions. Attach EC2FastLaunchFullAccess in the IAM console.
```

Penyebab

Pengguna atau peran IAM Anda tidak memiliki izin yang diperlukan untuk mengaktifkan Peluncuran Cepat EC2.

Solusi

Verifikasi bahwa prinsipal IAM Anda (pengguna atau peran) yang mengaktifkan Peluncuran Cepat EC2 memiliki `EC2FastLaunchFullAccess` kebijakan yang terlampir. Kebijakan ter AWS kelola ini memberikan akses penuh ke semua sumber daya Peluncuran Cepat EC2. Untuk melihat izin kebijakan ini, lihat kebijakan di Referensi EC2FastLaunchFullAccess Kebijakan AWS Terkelola.

# Mengubah kata sandi Administrator Windows untuk instans Amazon EC2 Anda

Jika Anda meluncurkan instance dari AMI AWS Windows, agen peluncuran yang sudah diinstal sebelumnya menetapkan kata sandi default sebagai berikut:

- Untuk Windows Server 2022 dan yang lebih baruEC2Launch v2, buat kata sandi default.
- Untuk Windows Server 2016 dan 2019, EC2Launch agen menghasilkan kata sandi default.
- Untuk Windows Server 2012 R2 dan sebelumnya, Layanan EC2Config menghasilkan kata sandi default.

> **Note**
>
> Untuk AMI Windows Server 2016 dan versi `Password never expires` lebih baru, dinonaktifkan untuk administrator lokal. Untuk versi AMI sebelum Windows Server 2016, `Password never expires` diaktifkan untuk administrator lokal.

## Mengubah kata sandi Administrator setelah terhubung

Saat Anda terhubung ke sebuah instans untuk pertama kalinya, kami menyarankan Anda untuk mengubah kata sandi Administrator dari nilai default-nya. Gunakan prosedur berikut untuk mengubah kata sandi Administrator untuk instans Windows.

> **Important**
>
> Simpan kata sandi baru di tempat yang aman. Anda tidak akan bisa mendapatkan kembali kata sandi baru menggunakan konsol Amazon EC2. Konsol hanya dapat mengambil kata sandi default. Jika Anda mencoba untuk menyambung ke instans menggunakan kata sandi default setelah mengubahnya, Anda akan mendapatkan pesan kesalahan "Kredensial Anda tidak berfungsi".

Untuk mengubah kata sandi Administrator lokal

1. Hubungkan ke instans dan buka prompt perintah.

2. Jalankan perintah berikut. Jika kata sandi baru Anda menyertakan karakter khusus, apit kata sandi dengan tanda kutip ganda.

```
net user Administrator "new_password"
```

3. Simpan kata sandi baru di tempat yang aman.

### Mengubah kata sandi yang hilang atau kedaluwarsa

Jika Anda kehilangan kata sandi atau kedaluwarsa, Anda dapat membuat kata sandi baru. Untuk prosedur pengaturan ulang kata sandi, lihat Setel ulang kata sandi administrator Windows untuk instans Windows Amazon EC2.

## Menambahkan komponen Windows Server opsional ke instans Windows Amazon EC2

Untuk mengakses dan menginstal komponen opsional, Anda harus menemukan snapshot EBS yang benar untuk versi Windows Server Anda, membuat volume dari snapshot, dan melampirkan volume ke instans Anda.

Sebelum Anda mulai

Gunakan alat baris perintah Konsol Manajemen AWS atau untuk mendapatkan ID instance dan Zona Ketersediaan instans Anda. Anda harus membuat volume EBS di Zona Ketersediaan yang sama dengan instans.

Gunakan salah satu prosedur berikut untuk menambahkan komponen Windows Server ke instance Anda.

Console

Untuk menambahkan komponen Windows ke instance Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Snapshot.
3. Dari bilah Filter, pilih Snapshot publik.
4. Tambahkan filter Alias Pemilik dan pilih amazon.
5. Tambahkan filter Deskripsi dan masukkan **`Windows`**.

6. Tekan Enter
7. Pilih snapshot yang sesuai dengan arsitektur sistem dan preferensi bahasa Anda. Misalnya, pilih Media Instalasi Bahasa Inggris Windows 2019 jika instans Anda menjalankan Windows Server 2019.
8. Pilih Tindakan, Buat volume dari snapshot.
9. Untuk Zona Ketersediaan, pilih Zona Ketersediaan yang cocok dengan instans Windows Anda. Pilih Tambahkan tanda dan masukkan **`Name`** untuk kunci tanda serta nama deskriptif untuk nilai tanda. Pilih Buat volume.
10. Di pesan volume Berhasil dibuat (spanduk hijau), pilih volume yang baru saja Anda buat.
11. Pilih Tindakan, Lampirkan Volume.
12. Dari instans, pilih ID instans.
13. Untuk Nama perangkat, masukkan nama perangkat untuk lampiran. Jika Anda memerlukan bantuan terkait nama perangkat, lihat Nama perangkat untuk volume di EC2 instans Amazon.
14. Pilih Lampirkan volume.
15. Hubungkan ke instans Anda dan sediakan volume. Untuk informasi selengkapnya, lihat Membuat volume Amazon EBS tersedia untuk digunakan di Panduan Pengguna Amazon EBS.

    > **Important**
    >
    > Jangan menginisialisasi volume.

16. Buka Panel Kontrol, Program dan Fitur. Pilih Aktifkan atau nonaktifkan fitur Windows. Jika Anda diminta untuk media instalasi, tentukan volume EBS dengan media instalasi.
17. (Opsional) Saat Anda selesai menggunakan media instalasi, Anda dapat melepaskan volume. Setelah Anda melepaskan volume, Anda dapat menghapusnya.

AWS CLI

Untuk menambahkan komponen Windows ke instance Anda

1. Gunakan perintah describe-snapshots dengan parameter `owner-ids` dan filter `description` untuk mendapatkan daftar snapshot media instalasi yang tersedia.

```
aws ec2 describe-snapshots \
    --owner-ids amazon \
```

```
    --filters Name=description,Values=Windows*
```

2. Dalam keluaran, catat ID snapshot yang cocok dengan arsitektur sistem dan preferensi bahasa Anda. Misalnya:

```
{
    "Snapshots": [
    ...
        {
            "OwnerAlias": "amazon",
            "Description": "Windows 2019 English Installation Media",
            "Encrypted": false,
            "VolumeId": "vol-be5eafcb",
            "State": "completed",
            "VolumeSize": 6,
            "Progress": "100%",
            "StartTime": "2019-10-25T20:00:47.000Z",
            "SnapshotId": "snap-22da283e",
            "OwnerId": "123456789012"
        },
    ...
   ]
}
```

3. Gunakan perintah create-volume untuk membuat volume dari snapshot. Tentukan Zona Ketersediaan yang sama dengan instans Anda.

```
aws ec2 create-volume \
    --snapshot-id snap-0abcdef1234567890 \
    --volume-type gp2 \
    --availability-zone us-east-1a
```

4. Pada keluaran, catat ID volume.

```
{
    "AvailabilityZone": "us-east-1a",
    "Encrypted": false,
    "VolumeType": "gp2",
    "VolumeId": "vol-01234567890abcdef",
    "State": "creating",
    "Iops": 100,
    "SnapshotId": "snap-0abcdef1234567890",
    "CreateTime": "2017-04-18T10:33:10.940Z",
```

```
    "Size": 6
}
```

5. Gunakan perintah attach-volume untuk melampirkan volume ke instans Anda.

```
aws ec2 attach-volume \
    --volume-id vol-0c98b37f30bcbc290 \
    --instance-id i-01474ef662b89480 \
    --device xvdg
```

6. Hubungkan ke instans Anda dan sediakan volume. Untuk informasi selengkapnya, lihat Membuat volume Amazon EBS tersedia untuk digunakan di Panduan Pengguna Amazon EBS.

> **Important**
>
> Jangan menginisialisasi volume.

7. Buka Panel Kontrol, Program dan Fitur. Pilih Aktifkan atau nonaktifkan fitur Windows. Jika Anda diminta untuk media instalasi, tentukan volume EBS dengan media instalasi.
8. (Opsional) Saat Anda selesai menggunakan media instalasi, gunakan perintah detach-volume untuk melepaskan volume dari instans Anda. Setelah Anda melepaskan volume, Anda dapat menggunakan perintah delete-volume untuk menghapus volume.

PowerShell

Untuk menambahkan komponen Windows ke instance Anda

1. Gunakan Get-EC2Snapshot cmdlet dengan `description` filter `Owner` dan untuk mendapatkan daftar snapshot media instalasi yang tersedia.

```
Get-EC2Snapshot `
    -Owner amazon `
    -Filter @{ Name="description"; Values="Windows*" }
```

2. Dalam keluaran, catat ID snapshot yang cocok dengan arsitektur sistem dan preferensi bahasa Anda. Contoh:

```
...
DataEncryptionKeyId :
```

```
Description         : Windows 2019 English Installation Media
Encrypted           : False
KmsKeyId            :
OwnerAlias          : amazon
OwnerId             : 123456789012
Progress            : 100%
SnapshotId          : snap-0abcdef1234567890
StartTime           : 10/25/2019 8:00:47 PM
State               : completed
StateMessage        :
Tags                : {}
VolumeId            : vol-01234567890abcdef
VolumeSize          : 6
...
```

3. Gunakan New-EC2Volume cmdlet untuk membuat volume dari snapshot. Tentukan Zona Ketersediaan yang sama dengan instans Anda.

```
New-EC2Volume `
    -AvailabilityZone us-east-1a `
    -VolumeType gp2 `
    -SnapshotId snap-0abcdef1234567890
```

4. Pada keluaran, catat ID volume.

```
Attachments      : {}
AvailabilityZone : us-east-1a
CreateTime       : 4/18/2017 10:50:25 AM
Encrypted        : False
Iops             : 100
KmsKeyId         :
Size             : 6
SnapshotId       : snap-0abcdef1234567890
State            : creating
Tags             : {}
VolumeId         : vol-01234567890abcdef
VolumeType       : gp2
```

5. Gunakan Add-EC2Volume cmdlet untuk melampirkan volume ke instance Anda.

```
Add-EC2Volume `
    -InstanceId i-1234567890abcdef0 `
    -VolumeId vol-01234567890abcdef `
```

```
    -Device xvdh
```

6. Hubungkan ke instans Anda dan sediakan volume. Untuk informasi selengkapnya, lihat Membuat volume Amazon EBS tersedia untuk digunakan di Panduan Pengguna Amazon EBS.

   > **Important**
   >
   > Jangan menginisialisasi volume.

7. Buka Panel Kontrol, Program dan Fitur. Pilih Aktifkan atau nonaktifkan fitur Windows. Jika Anda diminta untuk media instalasi, tentukan volume EBS dengan media instalasi.
8. (Opsional) Setelah selesai dengan media instalasi, gunakan Dismount-EC2Volume cmdlet untuk melepaskan volume dari instance Anda. Setelah Anda melepaskan volume, Anda dapat menggunakan Remove-EC2Volume cmdlet untuk menghapus volume.

## Instal Subsistem Windows untuk Linux pada instance EC2 Windows Anda

Subsistem Windows untuk Linux (WSL) adalah fitur dari Microsoft Windows. Dengan menginstal WSL pada instance EC2 Windows Anda, Anda dapat menjalankan alat Linux asli langsung di instans Windows Anda.

Ada dua versi Windows Subsystem for Linux (WSL): WSL 1 dan WSL 2. Untuk informasi selengkapnya, lihat Windows Subsistem untuk Dokumentasi Linux di situs web Microsoft.

Persyaratan

- Sistem operasi harus Windows Server 2019 atau yang lebih baru.
- Anda hanya dapat menginstal WSL 1 pada instans Windows tervirtualisasi (ukuran instans tidak mendukung `.metal` atau tidak mendukung virtualisasi bersarang).
- Anda dapat menginstal WSL 1 atau WSL 2 pada instans yang mendukung virtualisasi bersarang dan mengaktifkan Opsi `NestedVirtualization` CPU.
- Anda dapat menginstal WSL 1 atau WSL 2 pada instance bare metal (ukuran instansnya adalah `.metal`). Instans bare metal menyediakan dukungan yang diperlukan untuk virtualisasi bersarang secara default.

Untuk informasi selengkapnya tentang virtualisasi bersarang untuk EC2, lihat. Gunakan virtualisasi bersarang untuk menjalankan hypervisor di instans Amazon EC2

## Menginstal WSL pada instans Windows Anda

Untuk menginstal WSL 1

1. Instal WSL. Proses yang akan Anda gunakan tergantung pada versi Windows Server yang berjalan pada instance.
   - Windows Server 2022 dan yang lebih baru - Jalankan perintah instalasi standar berikut pada instance EC2 Anda.

     ```
     wsl --install --enable-wsl1 --no-launch
     ```

   - Windows Server 2019 - Aktifkan WSL dan kemudian instal WSL seperti yang dijelaskan dalam Instal WSL pada versi Windows Server sebelumnya di situs web Microsoft.
2. Mulai ulang instans EC2 Anda.

   ```
   shutdown -r -t 20
   ```

3. Untuk mengonfigurasi WSL untuk menggunakan WSL 1, jalankan perintah berikut pada instans Anda. Langkah ini diperlukan untuk instans tervirtualisasi (ukuran instance tidak `.metal` atau tidak dikonfigurasi untuk virtualisasi bersarang).

   ```
   wsl --set-default-version 1
   ```

4. Instal distribusi default.

   ```
   wsl --install
   ```

Untuk menginstal WSL 2 (.metal atau instance dengan virtualisasi bersarang diaktifkan)

Jalankan perintah instalasi standar berikut pada instance EC2 Anda. Secara default, WSL 2 diinstal.

```
wsl --install
```

# Utilitas pemecahan masalah Windows EC2

`EC2WinUtil`Driver menyediakan jenis dukungan pemecahan masalah berikut untuk instance Windows Anda.

Tumpukan panggilan crash

`EC2WinUtil`mengumpulkan informasi kerusakan dasar dari instans Anda dan menuliskannya ke konsol serial. Daftar berikut mencakup beberapa detail utama yang ditulis utilitas ke konsol.

- Identifikasi modul yang menghasilkan kesalahan.
- Kode kesalahan Windows yang terkait dengan acara tersebut.
- Jejak tumpukan panggilan terbaru.

Dengan rincian ini, Anda dapat melakukan analisis akar penyebab awal dan menentukan apakah analisis lebih lanjut diperlukan. Output ke konsol serial juga memungkinkan AWS untuk melacak tren crash untuk driver Amazon EC2, dan mendiagnosis kejadian crash skala besar.

> **Note**
>
> `EC2WinUtil`tidak mengumpulkan data pelanggan apa pun di tumpukan panggilan kerusakannya.

Untuk catatan rilis driver, lihat Riwayat versi Driver Utilitas Windows EC2

## Riwayat versi Driver Utilitas Windows EC2

Tabel berikut menunjukkan `EC2WinUtil` driver mana yang berjalan pada setiap versi Windows Server di Amazon EC2. Versi sebelumnya dari sistem operasi menggunakan driver yang sudah diinstal pada AWS Windows Server AMI dari mana instance diluncurkan. AMI yang dibagikan dengan Anda atau yang Anda berlangganan AWS Marketplace tidak memiliki driver yang sudah diinstal sebelumnya.

| Versi Windows Server | EC2WinUtil versi driver |
| --- | --- |
| Windows Server 2025 | versi terbaru |
| Windows Server 2022 | versi terbaru |

| Versi Windows Server | EC2WinUtil versi driver |
| --- | --- |
| Windows Server 2019 | versi terbaru |
| Windows Server 2016 | versi terbaru |

> **Note**
>
> Sebelum driver versi 3.0.0, `EC2WinUtil` driver tidak tersedia untuk diunduh untuk instalasi manual. Versi sebelumnya hanya tersedia sebagai driver prainstal untuk AMI AWS Windows.

Tabel berikut menjelaskan versi `EC2WinUtil` driver yang dirilis.

| Versi Package | Versi Driver | Detail | Tanggal rilis |
| --- | --- | --- | --- |
| 3.2.0 | 3.2.0 | Mekanisme logging konsol yang disempurnakan untuk digunakan dengan agen EC2launch v2. | Juni 2, 2026 |
| 3.1.1 | 3.1.1 | Peningkatan panjang tumpukan panggilan saat masuk ke output konsol. | Maret 3, 2026 |
| 3.1.0 | 3.1.0 | Peningkatan penanganan acara manajemen daya. | Februari 4, 2026 |
| 3.0.0 | 3.0.0 | Memodernisasi driver untuk Windows 10 dan menambahkan dukungan untuk instalasi sebagai driver primitif. | Juni 13, 2024 |
| 2.0.0 | 2.0.0 | Menambahkan dukungan untuk output pada port serial MMIO untuk jenis instance logam. Juga meningkat kan penguraian kerusakan dan memperbarui format output. | 23 Agustus 2018 |

| Versi Package | Versi Driver | Detail | Tanggal rilis |
|---|---|---|---|
| 1.0.1 | 1.0.1 | Mengubah nama driver `EC2WinUtil` karena konflik namespace dengan Amazon Inspector. Beberapa perbaikan bug disertakan. | 1 Maret 2018 |
| 1.0.0 | 1.0.0 | Pelepasan awal. Sopir itu awalnya dipanggil `AwsAgent`. | 28 November 2017 |

# Tingkatkan instance EC2 Windows ke versi Windows Server yang lebih baru

Jika sudah waktunya untuk memutakhirkan sistem operasi Windows Server pada instance Windows EC2 Anda dari versi sebelumnya, Anda dapat menggunakan salah satu metode berikut.

In-place meningkatkan

Upgrade di tempat beroperasi pada instans yang sudah ada. Hanya file sistem operasi yang terpengaruh selama proses ini, sementara pengaturan, peran server, dan data Anda dibiarkan utuh.

Migrasi (juga dikenal sebagai upgrade berdampingan)

Migrasi melibatkan pengambilan pengaturan, konfigurasi, dan data, dan memindahkannya ke sistem operasi yang lebih baru pada instance Windows EC2 baru. Anda dapat meluncurkan instans dari AMI Windows publik atau pribadi yang Anda berlangganan dari AWS Marketplace, atau AMI yang dibagikan dengan Anda. Anda juga dapat membuat AMI khusus dengan EC2 Image Builder. Lihat Panduan Pengguna Pembuat Gambar untuk informasi selengkapnya.

> **Note**
>
> AWS menyediakan satu set Amazon Machine Images (AMI) yang tersedia untuk umum untuk versi Windows Server yang berjalan pada instans EC2. AMI ini diperbarui setiap bulan. Untuk informasi tentang AMI Windows terbaru, lihat Refer AWS ensi AMI Windows.

Microsoft secara tradisional merekomendasikan migrasi ke versi Windows Server yang lebih baru daripada memutakhirkan di tempat. Migrasi dapat mengakibatkan lebih sedikit kesalahan atau masalah peningkatan, tetapi dapat memakan waktu lebih lama daripada peningkatan di tempat karena kebutuhan untuk menyediakan instans baru, merencanakan dan mem-port aplikasi, dan menyesuaikan pengaturan konfigurasi pada instans baru. Pemutakhiran langsung bisa lebih cepat, tetapi ketidaksesuaian perangkat lunak dapat menghasilkan kesalahan.

Daftar Isi


- Melakukan peningkatan di tempat pada instans EC2 Windows
- Gunakan runbook Otomasi untuk meng-upgrade instance EC2 Windows
- Memigrasikan instance EC2 Windows ke jenis Nitro-based instans
- Memecahkan masalah peningkatan sistem operasi pada instans EC2 Windows


# Melakukan peningkatan di tempat pada instans EC2 Windows

Sebelum Anda melakukan pemutakhiran langsung, Anda harus menentukan driver jaringan mana yang dijalankan instans. Driver jaringan PV memungkinkan Anda mengakses instans Anda menggunakan Desktop Jarak Jauh. Instans menggunakan driver AWS PV, Intel Network Adapter, atau Enhanced Networking. Untuk informasi selengkapnya, lihat Driver paravirtual untuk instans Windows.

## Sebelum Anda memulai pemutakhiran langsung

Selesaikan tugas berikut dan catat detail penting berikut sebelum Anda memulai pemutakhiran langsung.

- Baca dokumentasi Microsoft untuk memahami persyaratan pemutakhiran, masalah umum, dan batasan. Tinjau juga instruksi resmi untuk pemutakhiran.
  - Opsi Pemutakhiran untuk Windows Server 2012
  - Opsi Pemutakhiran untuk Windows Server 2012 R2
  - Opsi peningkatan dan konversi untuk Windows Server 2016 dan di atasnya
  - Peningkatan Windows Server
- Kami merekomendasikan untuk melakukan pemutakhiran sistem operasi pada instans dengan setidaknya 2 vCPU dan RAM 4 GB. Jika perlu, Anda dapat mengubah instans ke ukuran yang lebih besar dengan tipe yang sama (misalnya t2.small ke t2.large), melakukan pemutakhiran,

lalu mengubah ukurannya kembali ke ukuran aslinya. Jika Anda diminta untuk mempertahankan ukuran instans, Anda dapat memantau kemajuannya menggunakan tangkapan layar konsol instans. Untuk informasi selengkapnya, lihat Perubahan jenis instans Amazon EC2.

- Verifikasi bahwa volume root pada instans Windows Anda memiliki ruang disk yang cukup. Proses Penataan Windows mungkin tidak memperingatkan Anda tentang ruang disk yang tidak mencukupi. Untuk informasi tentang berapa banyak ruang disk yang diperlukan untuk memutakhirkan sistem operasi tertentu, lihat dokumentasi Microsoft. Jika volume tidak memiliki cukup ruang, volume dapat diperbesar. Untuk informasi selengkapnya, lihat Amazon EBS Elastic Volumes di Panduan Pengguna Amazon EBS.
- Tentukan jalur pemutakhiran Anda. Anda harus memutakhirkan sistem operasi ke arsitektur yang sama. Misalnya, Anda harus memutakhirkan sistem 32-bit ke sistem 32-bit. Windows Server 2008 R2 dan setelahnya hanya 64-bit.
- Nonaktifkan perangkat lunak antivirus dan anti-spyware serta firewall. Jenis perangkat lunak ini dapat bertentangan dengan proses peningkatan. Re-enable perangkat lunak antivirus dan anti-spyware dan firewall setelah peningkatan selesai.
- Perbarui ke driver terbaru seperti yang dijelaskan di Memigrasikan instance EC2 Windows ke jenis Nitro-based instans.
- Layanan Pembantu Pemutakhiran hanya mendukung instans yang menjalankan driver Citrix PV. Jika instans menjalankan driver Red Hat, Anda harus memutakhirkan driver tersebut secara manual terlebih dahulu.

## Tingkatkan instans di tempat dengan AWS PV, Intel Network Adapter, atau driver Jaringan yang Ditingkatkan

Gunakan prosedur berikut untuk memutakhirkan instans Windows Server menggunakan AWS PV, Adaptor Jaringan Intel, atau driver jaringan untuk Peningkatan Jaringan.

Untuk melakukan pemutakhiran langsung

1. Buat AMI dari sistem yang Anda rencanakan untuk dimutakhirkan, baik untuk tujuan pencadangan atau pengujian. Anda kemudian dapat melakukan pemutakhiran pada salinan untuk menyimulasikan lingkungan pengujian. Jika pemutakhiran selesai, Anda dapat mengalihkan lalu lintas ke instans ini dengan sedikit waktu henti. Jika pemutakhiran gagal, Anda dapat kembali ke cadangan. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.
2. Pastikan instans Windows Server Anda menggunakan driver jaringan terbaru.

a. Untuk memperbarui driver AWS PV Anda, lihatTingkatkan driver PV pada instans Windows EC2.

b. Untuk memperbarui driver ENA Anda, lihatInstal driver ENA pada instans EC2 Windows.

c. Untuk memperbarui driver Intel Anda, lihat Jaringan yang ditingkatkan dengan antarmuka Intel 82599 VF

3. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

4. Di panel navigasi, pilih Instans. Temukan instans tersebut. Catat ID instans dan Zona Ketersediaan untuk instans tersebut. Anda membutuhkan informasi ini nanti dalam prosedur ini.

5. Jika Anda memutakhirkan dari Windows Server 2012 atau 2012 R2 ke Windows Server 2016 atau yang lebih baru, lakukan hal berikut pada instance Anda sebelum melanjutkan.

   a. Hapus instalasi layanan EC2Config. Untuk informasi selengkapnya, lihat Administrasi Layanan Windows untuk agen EC2launch v2 dan EC2config.

   b. Instal EC2Launch v1 atau agen EC2Launch v2. Untuk informasi selengkapnya, lihat Gunakan agen EC2launch v1 untuk melakukan tugas selama peluncuran instance EC2 Windows dan Gunakan agen EC2launch v2 untuk melakukan tugas selama peluncuran instance EC2 Windows.

   c. Instal Agen AWS Systems Manager SSM. Untuk informasi selengkapnya, lihat Menginstal Agen SSM secara manual di Amazon EC2 untuk Windows Server di Panduan AWS Systems Manager Pengguna.

6. Buat volume baru dari snapshot media instalasi Windows Server.

   a. Di panel navigasi, di bagian Elastic Block Store, pilih Snapshot.

   b. Dari bilah filter, pilih Snapshot publik.

   c. Di bilah pencarian, tentukan filter berikut ini:

      - Pilih Alias Pemilik, lalu =, kemudian amazon.
      - Pilih Deskripsi, lalu mulai mengetik **`Windows`**. Pilih filter Windows yang cocok dengan arsitektur sistem dan preferensi bahasa yang Anda tingkatkan. Misalnya, pilih Media Instalasi Bahasa Inggris Windows 2019 untuk memutakhirkan ke Windows Server 2019.

   d. Pilih kotak centang di sebelah snapshot yang cocok dengan arsitektur sistem dan preferensi bahasa yang Anda tingkatkan, lalu pilih Tindakan , Buat volume dari snapshot.

   e. Di halaman Buat volume, pilih Zona Ketersediaan yang cocok dengan instans Windows Anda, dan pilih Buat volume.

7. Di *1234567890example* banner volume volume yang berhasil dibuat di bagian atas halaman, pilih ID volume yang baru saja Anda buat.
8. Pilih Tindakan, Lampirkan Volume.
9. Pada halaman Lampirkan volume, untuk Instans, pilih ID instans dari instans Windows Anda, lalu pilih Lampirkan volume.
10. Buat volume baru tersedia untuk digunakan dengan mengikuti langkah-langkah di Membuat volume Amazon EBS tersedia untuk digunakan.

> **Important**
>
> Jangan menginisialisasi disk karena melakukannya akan menghapus data yang ada.

11. Di Windows PowerShell, beralih ke drive volume baru. Mulailah pemutakhiran dengan membuka volume media instalasi yang Anda lampirkan pada instans.

    a. Jika Anda meningkatkan ke Windows Server 2016 atau lebih baru, jalankan perintah berikut:

    ```
    .\setup.exe /auto upgrade /dynamicupdate disable
    ```

    > **Note**
    >
    > Menjalankan setup.exe dengan opsi `/dynamicupdate` yang diatur ke nonaktif akan mencegah Windows menginstal pembaruan selama proses pemutakhiran Windows Server, karena menginstal pembaruan selama pemutakhiran dapat menyebabkan kegagalan. Anda dapat menginstal pembaruan dengan Windows Update setelah pemutakhiran selesai.

    Jika Anda meningkatkan ke Windows Server versi sebelumnya, jalankan perintah berikut:

    ```
    Sources\setup.exe
    ```

    b. Untuk Pilih sistem operasi yang ingin Anda instal, pilih opsi instalasi lengkap untuk instance Windows Server Anda, dan pilih Ber ikutnya.

    c. Untuk Jenis penginstalan apa yang Anda inginkan?, pilih Pemutakhiran.

    d. Selesaikan wizard.

Windows Server Setup menyalin dan memproses file. Setelah beberapa menit, sesi Remote Desktop Anda ditutup. Waktu yang diperlukan untuk memutakhirkan tergantung pada jumlah aplikasi dan peran server yang berjalan pada instans Windows Server Anda. Proses pemutakhiran dapat memakan waktu sedikitnya 40 menit atau beberapa jam. Instans mungkin gagal dalam satu atau beberapa pemeriksaan status selama proses pemutakhiran. Ketika peningkatan selesai, semua pemeriksaan status lulus. Anda dapat memeriksa log sistem untuk keluaran konsol atau menggunakan Amazon CloudWatch metrik untuk aktivitas disk dan CPU untuk menentukan apakah peningkatan sedang berlangsung.

> **Note**
>
> Jika memutakhirkan ke Windows Server 2019, setelah pemutakhiran selesai Anda dapat mengubah latar belakang desktop secara manual untuk menghapus nama sistem operasi sebelumnya jika diinginkan.

Jika instans belum lulus semua pemeriksaan status setelah beberapa jam, lihatMemecahkan masalah peningkatan sistem operasi pada instans EC2 Windows.

### Tugas pasca pemutakhiran

1. Masuk ke instans untuk menginisiasi pemutakhiran untuk .NET Framework dan boot ulang sistem saat diminta.
2. Jika Anda belum melakukannya pada langkah sebelumnya, instal agen EC2launch v1 atau EC2launch v2. Untuk informasi selengkapnya, lihat Gunakan agen EC2launch v1 untuk melakukan tugas selama peluncuran instance EC2 Windows dan Gunakan agen EC2launch v2 untuk melakukan tugas selama peluncuran instance EC2 Windows.
3. Jika Anda memutakhirkan ke Windows Server 2012 R2, kami sarankan Anda memutakhirkan driver PV ke driver AWS PV. Jika Anda memutakhirkan pada Nitro-based instance, kami sarankan Anda menginstal atau meningkatkan driver NVME dan ENA. Untuk informasi selengkapnya, lihat NVMe driver atau Mengaktifkan jaringan yang ditingkatkan di Windows.
4. Re-enable perangkat lunak antivirus dan anti-spyware dan firewall.

## Gunakan runbook Otomasi untuk meng-upgrade instance EC2 Windows

Anda dapat melakukan peningkatan otomatis instans Windows dan SQL Server Anda AWS dengan Runbook AWS Systems Manager Otomasi.

Daftar Isi


- Layanan terkait
- Opsi eksekusi
- Mutakhirkan Windows Server
- Mutakhirkan SQL Server


## Layanan terkait

AWS Layanan berikut digunakan dalam proses peningkatan otomatis:

- AWS Systems Manager. AWS Systems Manager adalah antarmuka yang kuat dan terpadu untuk mengelola AWS sumber daya Anda secara terpusat. Untuk informasi selengkapnya, silakan lihat Panduan Pengguna AWS Systems Manager.
- AWS Systems Manager Agen (SSM Agent) adalah perangkat lunak Amazon yang dapat diinstal dan dikonfigurasi pada instans Amazon EC2, server lokal, atau mesin virtual (VM). SSM Agent memungkinkan Systems Manager untuk memperbarui, mengelola, dan mengonfigurasi sumber daya ini. Agen memproses permintaan dari layanan Systems Manager di AWS Cloud, dan kemudian menjalankannya seperti yang ditentukan dalam permintaan. Untuk informasi selengkapnya, lihat Bekerja dengan SSM Agent dalam Panduan Pengguna AWS Systems Manager .
- AWS Systems Manager Buku run  book SSM. Runbook SSM menentukan tindakan yang dilakukan Systems Manager pada instans yang Anda kelola. Runbook SSM menggunakan JavaScript Object Notation (JSON) atau YAML, dan mereka menyertakan langkah-langkah dan parameter yang Anda tentukan. Topik ini menggunakan dua runbook SSM Systems Manager untuk otomatisasi. Untuk informasi selengkapnya, lihat Referensi runbook AWS Systems Manager Automation di Panduan Pengguna AWS Systems Manager .

## Opsi eksekusi

Saat Anda memilih Otomatisasi di konsol Systems Manager, pilih Jalankan. Setelah Anda memilih dokumen Otomatisasi, Anda akan diminta untuk memilih opsi eksekusi otomatisasi. Anda memilih dari opsi berikut. Dalam langkah-langkah untuk jalur yang disediakan dalam topik ini nanti, kami menggunakan opsi Eksekusi simpel.

Eksekusi sederhana

Pilih opsi ini jika Anda ingin memperbarui satu instans tetapi tidak ingin melalui setiap langkah otomasi untuk mengaudit hasil. Opsi ini dijelaskan lebih detail dalam langkah-langkah pemutakhiran yang mengikuti.

Kontrol tarif

Pilih opsi ini jika Anda ingin menerapkan pemutakhiran ke lebih dari satu instans. Anda menentukan pengaturan berikut.

- Parameter

  Pengaturan ini, yang juga diatur dalam Multi-Account dan Pengaturan Wilayah, menentukan bagaimana otomatisasi Anda bercabang.

- Target

  Pilih target yang ingin Anda terapkan otomatisasi. Pengaturan ini juga diatur dalam Multi-Account dan Pengaturan Wilayah.

- Nilai Parameter

  Gunakan nilai yang ditentukan dalam parameter dokumen otomatisasi.

- Grup Sumber Daya

  Dalam AWS, sumber daya adalah entitas yang dapat Anda kerjakan. Contohnya termasuk instans Amazon EC2, AWS CloudFormation stack, atau bucket Amazon S3. Jika Anda bekerja dengan beberapa sumber daya, mungkin berguna untuk mengelolanya sebagai grup daripada berpindah dari satu AWS layanan ke layanan lain untuk setiap tugas. Dalam beberapa kasus, Anda mungkin ingin mengelola sejumlah besar sumber daya terkait, seperti instans EC2 yang membentuk lapisan aplikasi. Dalam kasus ini, Anda mungkin perlu melakukan tindakan massal pada sumber daya ini sekaligus.

- Tanda

  Tag membantu Anda mengkategorikan AWS sumber daya Anda dengan cara yang berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Kategorisasi ini berguna jika Anda memiliki banyak sumber daya dengan tipe yang sama. Anda dapat dengan cepat mengidentifikasi sumber daya tertentu menggunakan tanda yang ditetapkan.

- Kontrol Tingkat

Kontrol Tarif juga diatur dalam Multi-Account dan Pengaturan Wilayah. Saat Anda menetapkan parameter kontrol tarif, Anda menentukan berapa banyak armada Anda yang akan menerapkan otomatisasi, baik berdasarkan jumlah target maupun persentase armada.

Multi-Account dan Wilayah

Selain parameter yang ditentukan di bawah Kontrol Tingkat yang juga digunakan dalam pengaturan Multi-Account dan Wilayah, ada dua pengaturan tambahan:

- Akun dan unit organisasi (OU)

  Tentukan beberapa akun tempat Anda ingin menjalankan otomatisasi.
- Wilayah AWS

  Tentukan beberapa Wilayah AWS tempat Anda ingin menjalankan otomatisasi.

Eksekusi manual

Opsi ini mirip dengan Eksekusi simpel, tetapi memungkinkan Anda untuk melangkah melalui setiap langkah otomatisasi dan mengaudit hasilnya.

## Mutakhirkan Windows Server

Runbook `AWSEC2-CloneInstanceAndUpgradeWindows` membuat Amazon Machine Image (AMI) dari instans Windows Server di akun Anda dan memutakhirkan AMI ini ke versi pilihan Anda yang didukung. Penyelesaian proses multilangkah ini dapat memakan waktu hingga dua jam.

Ada dua AMI yang disertakan dalam proses pemutakhiran otomatis:

- Instans yang sedang berjalan. AMI pertama adalah instans yang sedang berjalan, yang tidak dimutakhirkan. AMI ini digunakan untuk meluncurkan instans lain untuk menjalankan pemutakhiran langsung. Ketika prosesnya selesai, AMI ini dihapus dari akun, kecuali Anda secara khusus meminta untuk menyimpan instans aslinya. Pengaturan ini ditangani oleh parameter `KeepPreUpgradeImageBackUp` (nilai defaultnya adalah `false`, yang berarti AMI dihapus secara default).
- AMI yang dimutakhirkan. AMI ini adalah hasil dari proses otomasi.

Hasil akhirnya adalah satu AMI, yang merupakan instans AMI yang dimutakhirkan.

Saat pemutakhiran selesai, Anda dapat menguji fungsionalitas aplikasi Anda dengan meluncurkan AMI baru di Amazon VPC Anda. Setelah pengujian, dan sebelum Anda melakukan pemutakhiran lainnya, jadwalkan waktu henti aplikasi sebelum sepenuhnya beralih ke instans yang dimutakhirkan.

Prasyarat

Untuk mengotomatiskan pemutakhiran Windows Server Anda dengan dokumen AWS Systems Manager Otomasi, Anda harus melakukan tugas-tugas berikut:

- Buat peran IAM dengan kebijakan IAM yang ditentukan untuk memungkinkan Systems Manager melakukan tugas otomatisasi pada instans Amazon EC2 Anda dan memastikan Anda memenuhi prasyarat untuk menggunakan Systems Manager. Untuk informasi selengkapnya, lihat Membuat peran untuk mendelegasikan izin ke AWS layanan di Panduan AWS Identity and Access Management Pengguna.
- Pilih opsi bagaimana Anda ingin otomatisasi dijalankan. Opsi untuk eksekusi adalah Eksekusi Seder hana, Kontrol Tingkat, Multi-account dan Wilayah, dan Ek sekusi Manual. Untuk informasi selengkapnya tentang opsi ini, lihat Opsi eksekusi.
- Pastikan bahwa SSM Agent diinstal pada instans Anda. Untuk informasi selengkapnya, lihat Menginstal dan mengonfigurasi SSM Agent di instans Amazon EC2 untuk Windows Server.
- Windows PowerShell 3.0 atau yang lebih baru harus diinstal pada instance Anda.
- Untuk instans yang bergabung dengan domain Microsoft Active Directory, sebaiknya tentukan `SubnetId` yang tidak memiliki konektivitas ke kontroler domain Anda untuk membantu menghindari konflik nama host.
- Subnet instance harus memiliki konektivitas keluar ke internet, yang menyediakan akses Layanan AWS seperti Amazon S3 dan akses untuk mengunduh patch dari Microsoft. Persyaratan ini terpenuhi jika subnet adalah subnet publik dan instans memiliki alamat IP publik, atau jika subnet adalah subnet pribadi dengan rute yang mengirimkan lalu lintas internet ke perangkat NAT publik.
- Otomatisasi ini bekerja dengan instans yang menjalankan Windows Server 2008 R2, Windows Server 2012 R2, Windows Server 2016, dan Windows Server 2019.
- Verifikasi bahwa instans memiliki 20 GB ruang disk kosong di disk boot.
- Jika instance tidak menggunakan lisensi Windows yang disediakan oleh AWS, maka tentukan ID snapshot Amazon EBS yang menyertakan media instalasi Windows Server 2012 R2. Untuk melakukannya:
  1. Pastikan bahwa instans Amazon EC2 menjalankan Windows Server 2012 atau setelahnya.
  2. Buat volume Amazon EBS 6 GB di Zona Ketersediaan yang sama tempat instans berjalan. Lampirkan volume ke instans. Pasang, misalnya, sebagai drive D.

3. Buka menu konteks (klik kanan) untuk ISO dan pasang ke instance seperti, misalnya, drive E.
4. Salin konten ISO dari drive E:\ ke drive D:\
5. Buat snapshot Amazon EBS dengan volume 6 GB yang dibuat pada langkah 2 di atas.

### Batasan pemutakhiran Windows Server

Otomatisasi ini tidak mendukung peningkatan kontroler domain Windows, klaster, atau sistem operasi desktop Windows. Selain itu, otomatisasi ini tidak mendukung instans Amazon EC2 untuk Windows Server dengan peran berikut diinstal:

- Host Sesi Desktop Jarak Jauh (RDSH)
- Broker Koneksi Desktop Jarak Jauh (RDCB)
- Host Virtualisasi Desktop Jarak Jauh (RDVH)
- Akses Web Desktop Jarak Jauh (RDWA)

### Langkah-langkah untuk melakukan pemutakhiran otomatis Windows Server

Ikuti langkah-langkah ini untuk memutakhirkan instance Windows Server Anda menggunakan runbook AWSEC2-CloneInstanceAndUpgradeWindows otomatisasi.

1. Buka Systems Manager dari Konsol Manajemen.AWS
2. Dari panel navigasi kiri, pada Pengaturan, pilih Pengaturan.
3. Pilih Eksekusi otomatisasi.
4. Cari dokumen otomatisasi bernama `AWSEC2-CloneInstanceAndUpgradeWindows`.
5. Saat nama dokumen muncul, pilih nama itu. Saat Anda memilihnya, detail dokumen muncul.
6. Pilih Eksekusi otomatisasi untuk memasukkan parameter untuk dokumen ini. Biarkan Eksekusi sederhana dipilih di bagian atas halaman.
7. Masukkan parameter yang diminta berdasarkan panduan berikut.

   - `InstanceID`

     Tipe: String

     (Wajib) Instans yang menjalankan Windows Server 2008 R2, 2012 R2, 2016, atau 2019 dengan agen SSM terinstal.
   - `InstanceProfile`.

Tipe: String

(Wajib) Profil instans IAM. Ini adalah peran IAM yang digunakan untuk melakukan otomatisasi Manajer Sistem terhadap instans Amazon EC2 dan AWS AMI. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans EC2 di Panduan AWS Systems Manager Pengguna.

- `TargetWindowsVersion`

  Tipe: String

  (Wajib) Pilih versi Windows target.

- `SubnetId`

  Tipe: String

  (Wajib) Ini adalah subnet untuk proses pemutakhiran dan tempat instans EC2 sumber Anda berada. Verifikasi bahwa subnet memiliki konektivitas keluar ke AWS layanan, termasuk Amazon S3, dan juga ke Microsoft (untuk mengunduh patch).

- `KeepPreUpgradedBackUp`

  Tipe: String

  (Opsional) Jika parameter ini diatur ke `true`, otomatisasi mempertahankan gambar yang dibuat dari instans. Pengaturan default-nya adalah `false`.

- `RebootInstanceBeforeTakingImage`

  Tipe: String

  (Opsional) Default-nya adalah `false` (tanpa reboot). Jika parameter ini disetel ke`true`, Systems Manager melakukan boot ulang pada instans sebelum membuat AMI untuk peningkatan.

8. Setelah Anda memasukkan parameter, pilih Eksekusi. Saat otomatisasi dimulai, Anda dapat memantau kemajuan eksekusi.
9. Saat otomatisasi selesai, Anda akan melihat ID AMI. Anda dapat meluncurkan AMI untuk memverifikasi bahwa OS Windows telah dimutakhirkan.

> **Note**
>
> Tidak perlu otomatisasi untuk menjalankan semua langkah. Langkah-langkahnya bersyarat berdasarkan perilaku otomatisasi dan instans. Systems Manager mungkin melewatkan beberapa langkah yang tidak diperlukan.
> Selain itu, beberapa langkah mungkin kehabisan waktu. Systems Manager mencoba memutakhirkan dan menginstal semua patch terbaru. Namun, terkadang, tambalan waktu habis berdasarkan pengaturan batas waktu yang dapat ditentukan untuk langkah tertentu. Ketika ini terjadi, otomatisasi Systems Manager melanjutkan ke langkah berikutnya untuk memastikan bahwa OS internal dimutakhirkan ke versi Windows Server target.

10. Setelah otomatisasi selesai, Anda dapat meluncurkan instans Amazon EC2 menggunakan ID AMI untuk meninjau pemutakhiran Anda. Untuk informasi selengkapnya tentang cara membuat instans Amazon EC2 dari AWS AMI, lihat Bagaimana cara meluncurkan instans EC2 dari AMI kustom?

## Mutakhirkan SQL Server

Skrip AWSEC2-CloneInstanceAndUpgradeSQLServer membuat AMI dari instans Amazon EC2 yang menjalankan SQL Server di akun Anda, lalu memutakhirkan AMI ke SQL Server versi yang lebih baru. Penyelesaian proses multilangkah ini dapat memakan waktu hingga dua jam.

Dalam alur kerja ini, otomatisasi membuat AMI dari instans, lalu meluncurkan AMI baru di subnet yang Anda sediakan. Otomatisasi kemudian melakukan pemutakhiran langsung SQL Server. Setelah pemutakhiran selesai, otomatisasi membuat AMI baru sebelum menghentikan instans yang ditingkatkan.

Ada dua AMI yang disertakan dalam proses pemutakhiran otomatis:

- Instans yang sedang berjalan. AMI pertama adalah instans yang sedang berjalan, yang tidak dimutakhirkan. AMI ini digunakan untuk meluncurkan instans lain untuk menjalankan pemutakhiran langsung. Ketika prosesnya selesai, AMI ini dihapus dari akun, kecuali Anda secara khusus meminta untuk menyimpan instans aslinya. Pengaturan ini ditangani oleh parameter `KeepPreUpgradeImageBackUp` (nilai defaultnya adalah `false`, yang berarti AMI dihapus secara default).
- AMI yang dimutakhirkan. AMI ini adalah hasil dari proses otomasi.

Hasil akhirnya adalah satu AMI, yang merupakan instans AMI yang dimutakhirkan.

Saat pemutakhiran selesai, Anda dapat menguji fungsionalitas aplikasi Anda dengan meluncurkan AMI baru di Amazon VPC Anda. Setelah pengujian, dan sebelum Anda melakukan pemutakhiran lainnya, jadwalkan waktu henti aplikasi sebelum sepenuhnya beralih ke instans yang dimutakhirkan.

Prasyarat

Untuk mengotomatiskan pemutakhiran SQL Server Anda dengan dokumen AWS Systems Manager Otomasi, Anda harus melakukan tugas-tugas berikut:

- Buat peran IAM dengan kebijakan IAM yang ditentukan untuk memungkinkan Systems Manager melakukan tugas otomatisasi pada instans Amazon EC2 Anda dan memastikan Anda memenuhi prasyarat untuk menggunakan Systems Manager. Untuk informasi selengkapnya, silakan lihat Membuat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam AWS Identity and Access Management Panduan pengguna .
- Pilih opsi bagaimana Anda ingin otomatisasi dijalankan. Opsi untuk eksekusi adalah Eksekusi Seder hana, Kontrol Tingkat, Multi-account dan Wilayah, dan Ek sekusi Manual. Untuk informasi selengkapnya tentang opsi ini, lihat Opsi eksekusi.
- Instans Amazon EC2 harus menggunakan Windows Server 2008 R2 atau setelahnya dan SQL Server 2008 atau setelahnya.
- Pastikan bahwa SSM Agent diinstal pada instans Anda. Untuk informasi selengkapnya, lihat Bekerja dengan SSM Agent di instans Amazon EC2 untuk Windows Server.
- Verifikasi bahwa instans memiliki cukup ruang disk:
  - Jika Anda memutakhirkan dari Windows Server 2008 R2 ke 2012 R2, atau dari Windows Server 2012 R2 ke sistem operasi yang lebih baru, verifikasi bahwa Anda memiliki 20 GB ruang disk kosong di disk boot instans.
  - Jika Anda memutakhirkan dari Windows Server 2008 R2 ke 2016 atau versi lebih baru, verifikasi bahwa instans memiliki 40 GB ruang disk kosong di disk boot instans.
- Untuk instans yang menggunakan versi SQL Server Bawa Lisensi Sendiri (BYOL), prasyarat tambahan berikut ini berlaku:
  - Berikan ID snapshot Amazon EBS yang menyertakan media instalasi SQL Server target. Untuk melakukannya:
    1. Verifikasi bahwa instans Amazon EC2 menjalankan Windows Server 2008 R2 atau setelahnya.

2. Buat volume Amazon EBS 6 GB di Zona Ketersediaan yang sama tempat instans berjalan. Lampirkan volume ke instans. Pasang, misalnya, sebagai drive D.
3. Buka menu konteks (klik kanan) untuk ISO dan pasang ke instance seperti, misalnya, drive E.
4. Salin konten ISO dari drive E:\ ke drive D:\
5. Buat snapshot Amazon EBS dengan volume 6 GB yang dibuat pada langkah 2.

### Batasan pemutakhiran otomatis SQL Server

Batasan berikut berlaku saat menggunakan AWSEC2-CloneInstanceAndUpgradeSQLServer runbook untuk melakukan peningkatan otomatis:

- Pemutakhiran dapat dilakukan hanya pada SQL Server menggunakan autentikasi Windows.
- Verifikasi bahwa tidak ada pembaruan patch keamanan yang tertunda pada instans. Buka Panel Kontrol, lalu pilih Periksa pembaruan.
- Penerapan SQL Server dalam mode HA dan mirroring tidak didukung.

### Langkah-langkah untuk melakukan pemutakhiran otomatis SQL Server

Ikuti langkah-langkah ini untuk meng-upgrade SQL Server Anda menggunakan runbook AWSEC2-CloneInstanceAndUpgradeSQLServer otomatisasi.

1. Jika Anda belum melakukannya, unduh file.iso SQL Server 2016 dan pasang ke server sumber.
2. Setelah file .iso dipasang, salin semua file komponen dan letakkan di volume apa pun pilihan Anda.
3. Ambil snapshot volume Amazon EBS dan salin ID snapshot ke clipboard untuk digunakan nanti. Untuk informasi selengkapnya, lihat Membuat snapshot Amazon EBS di Panduan Pengguna Amazon EBS.
4. Lampirkan profil instans ke instans sumber Amazon EC2. Ini memungkinkan Manajer Sistem untuk berkomunikasi dengan instance EC2 dan menjalankan perintah di atasnya setelah ditambahkan ke AWS Systems Manager layanan. Untuk contoh ini, kami menamai peran tersebut `SSM-EC2-Profile-Role` dengan kebijakan `AmazonSSMManagedInstanceCore` yang dilampirkan pada peran tersebut.
5. Di AWS Systems Manager konsol, di panel navigasi kiri, pilih Instans Ter kelola. Verifikasi bahwa instans EC2 Anda ada dalam daftar instans terkelola. Jika Anda tidak melihat instans

Anda setelah beberapa menit, lihat Di Mana Instans Saya? di Panduan Pengguna AWS Systems Manager .

6. Dari panel navigasi kiri, pada Manajemen Perubahan, pilih Otomatisasi.
7. Pilih Eksekusi otomatisasi.
8. Cari dokumen otomatisasi bernama `AWSEC2-CloneInstanceAndUpgradeSQLServer`.
9. Pilih tombol dokumen `AWSEC2-CloneInstanceAndUpgradeSQLServer` SSM, lalu pilih Berikutnya.
10. Pastikan opsi Eksekusi simpel dipilih.
11. Masukkan parameter yang diminta berdasarkan panduan berikut.

    - `InstanceId`

      Tipe: String

      (Wajib) Instans yang menjalankan SQL Server 2008 R2 (atau setelahnya).

    - `IamInstanceProfile`

      Tipe: String

      (Wajib) Profil instans IAM.

    - `SQLServerSnapshotId`

      Tipe: String

      (Wajib) ID Snapshot untuk media penginstalan SQL Server target. Parameter ini tidak diperlukan untuk instans yang disertakan lisensi SQL Server.

    - `SubnetId`

      Tipe: String

      (Wajib) Ini adalah subnet untuk proses pemutakhiran dan tempat instans EC2 sumber Anda berada. Verifikasi bahwa subnet memiliki konektivitas keluar ke AWS layanan, termasuk Amazon S3, dan juga ke Microsoft (untuk mengunduh patch).

    - `KeepPreUpgradedBackUp`

      Tipe: String

(Opsional) Jika parameter ini diatur ke `true`, otomatisasi mempertahankan gambar yang dibuat dari instans. Pengaturan default-nya adalah `false`.

- `RebootInstanceBeforeTakingImage`

  Tipe: String

  (Opsional) Default-nya adalah `false` (tanpa reboot). Jika parameter ini atur ke `true`, Systems Manager melakukan boot ulang pada instans sebelum membuat AMI untuk peningkatan.

- `TargetSQLVersion`

  Tipe: String

  (Opsional) Versi SQL Server target. Nilai default-nya `2016`.

12. Setelah Anda memasukkan parameter, pilih Eksekusi. Saat otomatisasi dimulai, Anda dapat memantau kemajuan eksekusi.
13. Ketika Status eksekusi menunjukkan Berhasil, luaskan Output untuk melihat informasi AMI. Anda dapat menggunakan AMI ID untuk meluncurkan instans SQL Server untuk VPC pilihan Anda.
14. Buka konsol Amazon EC2. Di panel navigasi kiri, pilih AMI. Anda akan melihat AMI baru.
15. Untuk memverifikasi bahwa versi SQL Server yang baru telah berhasil diinstal, pilih AMI baru dan pilih Luncurkan.
16. Pilih tipe instans yang Anda inginkan untuk AMI, VPC dan subnet yang ingin Anda deploy, dan penyimpanan yang ingin Anda gunakan. Karena Anda meluncurkan instans baru dari AMI, volume disajikan kepada Anda sebagai opsi untuk disertakan dalam instans EC2 baru yang Anda luncurkan. Anda dapat menghapus salah satu volume ini, atau Anda dapat menambahkan volume.
17. Tambahkan tanda untuk membantu Anda mengidentifikasi instans Anda.
18. Tambahkan grup keamanan atau grup ke instans.
19. Pilih Luncurkan Instans.
20. Pilih nama tanda untuk instans tersebut dan pilih Hubungkan di bawah menu tarik-turun Tindakan.
21. Verifikasi bahwa versi SQL Server baru adalah mesin basis data pada instans baru.

# Memigrasikan instance EC2 Windows ke jenis Nitro-based instans

AMI AWS Windows dikonfigurasi dengan pengaturan default yang digunakan oleh media instalasi Microsoft, dengan beberapa penyesuaian. Penyesuaian mencakup driver dan konfigurasi yang mendukung Nitro-based instance, seperti M5 dan C5.

Saat bermigrasi dari Xen-based Nitro-based instance ke instance, termasuk instans bare metal, sebaiknya ikuti langkah-langkah dalam topik ini dalam kasus berikut:

- Jika Anda meluncurkan instans dari AMI Windows kustom
- Jika Anda meluncurkan instans dari AMI Windows yang disediakan oleh Amazon yang dibuat sebelum Agustus 2018

Atau, Anda dapat menggunakan dokumen otomatisasi `AWSSupport-UpgradeWindowsAWSDrivers` untuk mengotomatisasi prosedur yang dijelaskan di Bagian 1, Bagian 2, dan Bagian 3. Jika Anda memilih untuk menggunakan prosedur otomatis, lihat (Alternatif) Tingkatkan AWS Driver PV, ENA, dan NVMe menggunakan AWS Systems Manager, lalu lanjutkan dengan Bagian 4 dan Bagian 5.

Untuk informasi selengkapnya, lihat Pembaruan Amazon EC2 - Tipe Instans Tambahan, Nitro System, dan Opsi CPU.

> **Note**
>
> Prosedur migrasi berikut dapat dilakukan pada Windows Server versi 2016 dan yang lebih baru. Versi sistem operasi sebelumnya yang telah mencapai akhir masa pakai tidak diuji, dan mungkin tidak kompatibel dengan jenis instans terbaru.
> Untuk memigrasikan instans Linux, lihatthe section called “Perubahan jenis instans”.

Daftar Isi


- Bagian 1: Instal dan tingkatkan AWS Pengemudi PV
- Bagian 2: Instal dan mutakhirkan ENA
- Bagian 3: Tingkatkan AWS NVMe driver
- Bagian 4: Perbarui EC2Config dan EC2Launch
- Bagian 5: Instal driver port serial untuk instans bare metal

- Bagian 6: Perbarui pengaturan manajemen daya
- Bagian 7: Perbarui driver chipset Intel untuk tipe instans baru
- (Alternatif) Tingkatkan AWS Driver PV, ENA, dan NVMe menggunakan AWS Systems Manager

Sebelum Anda memulai

Prosedur ini mengasumsikan bahwa Anda memiliki Xen-based instance, seperti M4 atau C4, dan Anda bermigrasi ke instance. Nitro-based

Anda harus menggunakan PowerShell versi 3.0 atau yang lebih baru untuk berhasil melakukan upgrade.

> **Note**
>
> Saat melakukan migrasi, IP statis atau pengaturan jaringan DNS khusus pada kartu antarmuka jaringan yang ada mungkin hilang karena instans akan secara default ke perangkat Adaptor Jaringan yang Ditingkatkan baru.

Sebelum mengikuti langkah-langkah dalam prosedur ini, kami menyarankan Anda untuk membuat cadangan instans. Dari konsol EC2, pilih instans yang memerlukan migrasi, buka menu konteks (klik kanan), dan pilih Status Instans, Berhenti.

> **Warning**
>
> Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menyimpan data dalam volume penyimpanan instan, pastikan Anda mencadangkan data ke penyimpanan persisten.

Buka menu konteks (klik kanan) untuk instans di konsol EC2, pilih Gambar, lalu pilih Buat Gambar.

> **Note**
>
> Bagian 4 dan 5 dari instruksi ini dapat diselesaikan setelah Anda memigrasi atau mengubah jenis instans. Namun, sebaiknya Anda menyelesaikannya sebelum bermigrasi, terutama jika Anda bermigrasi ke jenis instans bare metal.

## Bagian 1: Instal dan tingkatkan AWS Pengemudi PV

Meskipun AWS driver PV tidak digunakan dalam sistem Nitro, Anda masih harus memutakhirkannya jika Anda menggunakan versi sebelumnya dari Citrix PV atau AWS PV. Driver AWS PV terbaru menyelesaikan bug di versi driver sebelumnya yang mungkin muncul saat Anda menggunakan sistem Nitro, atau jika Anda perlu bermigrasi kembali ke Xen-based instance. Sebagai praktik terbaik, kami sarankan selalu memperbarui ke driver terbaru untuk instans Windows aktif AWS.

Gunakan prosedur berikut untuk melakukan upgrade driver AWS PV di tempat, atau untuk memutakhirkan dari driver Citrix PV ke driver AWS PV pada Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016, atau Windows Server 2019. Untuk informasi selengkapnya, lihat Tingkatkan driver PV pada instans Windows EC2.

Untuk memutakhirkan Kontroler Domain, lihat Tingkatkan pengontrol domain (AWS Peningkatan PV).

Untuk melakukan peningkatan atau ke AWS Pengemudi PV

1. Hubungkan ke instans menggunakan Desktop Jarak Jauh dan persiapkan instans untuk pemutakhiran. Buat semua disk non-sistem offline sebelum Anda melakukan pemutakhiran. Jika Anda melakukan pembaruan driver AWS PV di tempat, langkah ini tidak diperlukan. Setel layanan yang tidak penting ke Pengaktifan manual di konsol Layanan.
2. Unduh paket driver terbaru ke instans.
3. Ekstrak isi folder dan jalankan `AWSPVDriverSetup.msi`.

Setelah menjalankan MSI, instans secara otomatis melakukan boot ulang dan memutakhirkan driver. Instans mungkin tidak tersedia hingga 15 menit.

Setelah peningkatan selesai dan instans lulus pada pemeriksaan kondisi di konsol Amazon EC2, hubungkan ke instans menggunakan Desktop Jarak Jauh dan pastikan driver baru terinstal. Di Pengelola Perangkat, di bawah Kontroler Penyimpanan, temukan Adaptor Host Penyimpanan AWS PV. Pastikan versi driver sama dengan versi terbaru yang terdaftar pada tabel Riwayat Versi Driver. Untuk informasi selengkapnya, lihat AWS Riwayat paket driver PV.

## Bagian 2: Instal dan mutakhirkan ENA

Mutakhirkan ke driver Adaptor Jaringan Elastis terbaru untuk memastikan bahwa semua fitur jaringan didukung. Jika Anda meluncurkan instans dan jaringan yang ditingkatkan belum diaktifkan, Anda harus mengunduh dan menginstal driver adaptor jaringan yang diperlukan pada instans Anda.

Kemudian, setel atribut instans enaSupport untuk mengaktifkan jaringan yang ditingkatkan. Anda hanya dapat mengaktifkan atribut ini pada tipe instans yang didukung dan hanya jika driver ENA diinstal. Untuk informasi selengkapnya, lihat Aktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda.

1. Unduh driver terbaru ke instans. Jika Anda memerlukan versi driver sebelumnya, lihatRIWAYAT VERSI DRIVER ENA Windows.
2. Ekstrak arsip zip.
3. Instal driver dengan menjalankan `install.ps1` PowerShell skrip dari folder yang diekstrak.

> **Note**
>
> Untuk menghindari kesalahan penginstalan, jalankan`install.ps1` skrip sebagai administrator.

4. Periksa apakah AMI Anda telah mengaktifkan enaSupport. Jika tidak, lanjutkan dengan mengikuti dokumentasi di Aktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda.

### Bagian 3: Tingkatkan AWS NVMe driver

AWS Driver NVMe digunakan untuk berinteraksi dengan volume penyimpanan instans Amazon EBS dan SSD yang diekspos sebagai perangkat blok NVMe dalam sistem Nitro untuk kinerja yang lebih baik.

> **Important**
>
> Instruksi berikut dimodifikasi khusus untuk saat Anda menginstal atau memutakhirkan AWS NVMe pada Xen-based instans dengan maksud untuk memigrasikan instance ke Nitro-based instance.

1. Unduh paket driver terbaru ke instans.

   Jika Anda memerlukan versi driver sebelumnya, lihat AWS Rilis driver NVMe Windows versi yang didukung.
2. Ekstrak arsip zip.

3. Instal driver seperti yang dijelaskan di`Readme.txt`.
4. Buka PowerShell dan jalankan perintah berikut:

```
PS C:\> start rundll32.exe sppnp.dll,Sysprep_Generalize_Pnp -wait
```

> **Note**
>
> Untuk menerapkan perintah, Anda harus menjalankan PowerShell sesi sebagai administrator. PowerShell Versi (x86) akan menghasilkan kesalahan.
> Perintah ini hanya menjalankan sysprep pada driver perangkat. Itu tidak menjalankan persiapan sysprep lengkap.

5. Untuk Windows Server 2008 R2 dan Windows Server 2012, matikan instance, ubah jenis instance dan mulai instance, lalu lanjutkan ke Bagian 4. Jika Anda memulai kembali instance pada tipe Xen-based instance sebelum bermigrasi ke jenis Nitro-based instance, instans tidak akan boot. Untuk AMI Windows lain yang didukung, Anda dapat mengubah tipe instans kapan saja setelah sysprep perangkat.

## Bagian 4: Perbarui EC2Config dan EC2Launch

Untuk instans Windows, utilitas EC2Config dan EC2Launch terbaru menyediakan fungsionalitas dan informasi tambahan saat dijalankan pada sistem Nitro, termasuk pada EC2 Bare Metal. Secara default, file EC2Config sudah disertakan dalam AMI sebelum Windows Server 2016. EC2Launch menggantikan EC2Config di AMI Windows Server 2016 dan setelahnya.

Ketika layanan EC2Config dan EC2Launch diperbarui, AMI Windows baru dari AWS menyertakan versi layanan terbaru. Namun demikian, Anda harus memperbarui AMI Windows dan instans Anda sendiri dengan EC2Config dan EC2Launch versi terbaru.

Untuk menginstal atau memperbarui EC2Config

1. Unduh dan unzip Penginstal EC2Config.
2. Jalankan `EC2Install.exe`. Untuk daftar lengkap opsi, jalankan `EC2Install` dengan opsi `/?`. Secara default, penyiapan menampilkan perintah. Untuk menjalankan perintah tanpa prompt, gunakan opsi `/quiet`.

Untuk informasi selengkapnya, lihat Menginstal EC2Config versi terbaru.

Untuk menginstal atau memperbarui EC2Launch

1. Jika Anda sudah menginstal dan mengonfigurasi EC2Launch pada sebuah instans, buat cadangan file konfigurasi EC2Launch. Proses penginstalan tidak menyimpan perubahan dalam file ini. Secara default, file terletak di direktori `C:\ProgramData\Amazon\EC2-Windows\Launch\Config`.
2. Unduh EC2-Windows-Launch.zip ke direktori pada instance.
3. Unduh install.ps1 ke direktori yang sama tempat Anda mengunduh `EC2-Windows-Launch.zip`.
4. Jalankan `install.ps1`.

> **Note**
>
> Untuk menghindari kesalahan penginstalan, jalankan skrip `install.ps1` sebagai administrator.

5. Jika Anda membuat cadangan file konfigurasi EC2Launch, salin file ke direktori `C:\ProgramData\Amazon\EC2-Windows\Launch\Config`.

Untuk informasi selengkapnya, lihat Gunakan agen EC2launch v1 untuk melakukan tugas selama peluncuran instance EC2 Windows.

## Bagian 5: Instal driver port serial untuk instans bare metal

Jenis `i3.metal` instance menggunakan perangkat PCI-based serial daripada perangkat serial I/O berbasis port. AMI Windows terbaru secara otomatis menggunakan perangkat PCI-based serial dan menginstal driver port serial. Jika Anda tidak menggunakan instance yang diluncurkan dari AMI Amazon-provided Windows tanggal 2018.04.11 atau yang lebih baru, Anda harus menginstal Driver Port Serial untuk mengaktifkan perangkat serial untuk fitur EC2 seperti Pembuatan Kata Sandi dan Output Konsol. Utilitas EC2Config dan EC2Launch terbaru juga mendukung i3.metal dan menyediakan fungsionalitas tambahan. Ikuti langkah-langkah di Bagian 4, jika Anda belum melakukannya.

Untuk menginstal driver port serial

1. Unduh paket driver serial ke instans.
2. Ekstrak konten folder, buka menu konteks (klik kanan) untuk, `aws_ser.INF`dan pilih instal.

3. Pilih Oke.

## Bagian 6: Perbarui pengaturan manajemen daya

Pembaruan berikut untuk pengaturan manajemen daya mengatur tampilan ke tidak pernah mati, yang memungkinkan pematian terkontrol OS pada sistem Nitro. Semua AMI Windows yang disediakan oleh Amazon sejak 28.11.2018 sudah memiliki konfigurasi default ini.

1. Buka command prompt atau PowerShell sesi.
2. Jalankan perintah berikut:

```
powercfg /setacvalueindex 381b4222-f694-41f0-9685-ff5bb260df2e 7516b95f-
f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
powercfg /setacvalueindex 8c5e7fda-e8bf-4a96-9a85-a6e23a8c635c 7516b95f-
f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
powercfg /setacvalueindex a1841308-3541-4fab-bc81-f71556f20b4a 7516b95f-
f776-4464-8c53-06167f40cc99 3c0bc021-c8a8-4e07-a973-6b14cbcb2b7e 0
```

## Bagian 7: Perbarui driver chipset Intel untuk tipe instans baru

Tipe instans, `u-6tb1.metalu-9tb1.metal`, dan `u-12tb1.metal` menggunakan perangkat keras yang memerlukan driver chipset yang sebelumnya tidak diinstal pada AMI Windows. Jika Anda tidak menggunakan instance yang diluncurkan dari AMI Amazon-provided Windows tanggal 2018.11.19 atau yang lebih baru, Anda harus menginstal driver menggunakan Intel Chipset INF Utility.

Untuk menginstal driver chipset

1. Utilitas Chipset INF ke instance.
2. Ekstrak file.
3. Jalankan `SetupChipset.exe`.
4. Terima perjanjian lisensi perangkat lunak Intel dan instal driver chipset.
5. Boot ulang instans.

## (Alternatif) Tingkatkan AWS Driver PV, ENA, dan NVMe menggunakan AWS Systems Manager

Dokumen otomatisasi `AWSSupport-UpgradeWindowsAWSDrivers` mengotomatisasi langkah-langkah yang dijelaskan di Bagian 1, Bagian 2, dan Bagian 3. Metode ini juga dapat memperbaiki instans di mana pemutakhiran driver gagal.

Dok `AWSSupport-UpgradeWindowsAWSDrivers` umen otomatisasi meningkatkan atau memperbaiki penyimpanan dan AWS driver jaringan pada instance EC2 yang ditentukan. Dokumen mencoba menginstal versi terbaru AWS driver online dengan memanggil AWS Systems Manager Agen (Agen SSM). Jika Agen SSM tidak dapat dihubungi, dokumen dapat melakukan instalasi offline AWS driver jika diminta secara eksplisit.

> **Note**
>
> Prosedur ini akan gagal pada kontroler domain. Untuk memperbarui driver pada kontroler domain, lihat Tingkatkan pengontrol domain (AWS Peningkatan PV).

Untuk meng-upgrade secara otomatis AWS Driver PV, ENA, dan NVMe menggunakan AWS Systems Manager

1. Buka konsol Manajer Sistem di https://console.aws.amazon.com/systems-manager.
2. Pilih Otomasi, Eksekusi Otomasi.
3. Cari dan kemudian pilih dokumen AWSSupport-UpgradeWindowsAWSDrivers otomatisasi, lalu pilih Jalankan otom atisasi.
4. Di bagian Parameter Input, konfigurasikan opsi berikut:

   ID Instans

   Masukkan ID unik dari instance yang akan ditingkatkan.

   AllowOffline

   (Opsional) Pilih salah satu opsi berikut:

   - `True` — Pilih opsi ini untuk melakukan penginstalan offline. Instans dihentikan dan dimulai ulang selama proses pemutakhiran.

> **Warning**
>
> Ketika Anda menghentikan instans, data pada setiap volume penyimpanan instans akan dihapus. Untuk menyimpan data dalam volume penyimpanan instan, pastikan Anda mencadangkan data ke penyimpanan persisten.

- `False` — (Default) Untuk melakukan penginstalan online, biarkan opsi ini dipilih. Instans dimulai ulang selama proses pemutakhiran.

> **Important**
>
> Pemutakhiran online dan offline membuat AMI sebelum mencoba operasi pemutakhiran. AMI tetap ada setelah otomatisasi selesai. Amankan akses Anda ke AMI, atau hapus jika tidak lagi diperlukan.

SubnetId

(Opsional) Masukkan salah satu nilai berikut:

- `SelectedInstanceSubnet` — (Default) Proses pemutakhiran meluncurkan instans helper ke subnet yang sama dengan instans yang akan dimutakhirkan. Subnet harus mengizinkan komunikasi ke titik akhir Systems Manager (`ssm.*`).
- `CreateNewVPC` — Proses pemutakhiran meluncurkan instans pembantu menjadi VPC baru. Gunakan opsi ini jika Anda tidak yakin apakah subnet instans target mengizinkan komunikasi ke titik akhir `ssm.*`. Pengguna Anda harus memiliki izin untuk membuat VPC.
- ID subnet tertentu — Tentukan ID subnet tertentu yang akan digunakan untuk meluncurkan instans helper. Subnet harus berada di Zona Ketersediaan yang sama dengan instans yang akan dimutakhirkan, dan harus mengizinkan komunikasi dengan titik akhir `ssm.*`.

5. Pilih Eksekusi.
6. Izinkan pemutakhiran selesai. Diperlukan waktu hingga 10 menit untuk menyelesaikan pemutakhiran secara online, dan hingga 25 menit untuk menyelesaikan pemutakhiran secara offline.

# Memecahkan masalah peningkatan sistem operasi pada instans EC2 Windows

AWS menyediakan dukungan upgrade untuk masalah atau masalah dengan Layanan Pembantu Upgrade, AWS utilitas yang membantu Anda melakukan peningkatan di tempat yang melibatkan driver Citrix PV.

Setelah pemutakhiran, instans mungkin untuk sementara mengalami penggunaan CPU yang lebih tinggi dari rata-rata sementara layanan .NET Runtime Optimization mengoptimalkan kerangka kerja .NET. Ini adalah perilaku yang diharapkan.

Jika instance belum lulus semua pemeriksaan status setelah beberapa jam, periksa yang berikut ini.

- Jika Anda memutakhirkan ke Windows Server 2008 dan semua pemeriksaan status gagal setelah beberapa jam, pemutakhiran mungkin gagal dan menampilkan prompt untuk  Klik OK  untuk mengonfirmasi pemutaran kembali. Karena konsol tidak dapat diakses pada status ini, tombol tersebut tidak dapat diklik Untuk menyiasatinya, lakukan reboot melalui konsol atau API Amazon EC2. Boot ulang membutuhkan waktu sepuluh menit atau lebih untuk memulai. Instans mungkin tersedia setelah 25 menit.
- Hapus aplikasi atau peran server dari server dan coba lagi.

Jika instance tidak lulus semua pemeriksaan status setelah menghapus aplikasi atau peran server dari server, lakukan hal berikut.

- Hentikan instans dan lampirkan volume root ke instans lain. Untuk informasi selengkapnya, lihat penjelasan cara menghentikan dan melampirkan volume root ke instans lain di “Menunggu layanan metadata”.
- Menganalisis file log Windows Setup dan log peristiwa untuk kegagalan.

Untuk isu atau masalah lain tekait pemutakhiran atau migrasi sistem operasi, sebaiknya tinjau artikel yang tercantum di Sebelum Anda memulai pemutakhiran langsung.

# Tutorial: Hubungkan instans Amazon EC2 ke basis data Amazon RDS

## Tujuan Tutorial

Tujuan dari tutorial ini adalah untuk mempelajari cara mengonfigurasi koneksi aman antara instans Amazon EC2 dan basis data Amazon RDS dengan menggunakan Konsol Manajemen AWS.

Ada berbagai opsi untuk mengonfigurasi koneksi. Dalam tutorial ini, kami mengeksplorasi tiga opsi berikut ini:

- Opsi 1: Secara otomatis menghubungkan instance ke database RDS menggunakan konsol EC2

  Gunakan fitur koneksi otomatis di konsol EC2 untuk secara otomatis mengonfigurasi koneksi antara instans EC2 dan basis data RDS Anda untuk memungkinkan lalu lintas antara instans EC2 dan basis data RDS.
- Opsi 2: Secara otomatis menghubungkan instance ke database RDS menggunakan konsol RDS

  Gunakan fitur koneksi otomatis di konsol RDS untuk secara otomatis mengonfigurasi koneksi antara instans EC2 dan basis data RDS Anda untuk memungkinkan lalu lintas antara instans EC2 dan basis data RDS.
- Opsi 3: Menghubungkan instance secara manual ke database RDS dengan membuat grup keamanan

  Konfigurasikan koneksi antara instans EC2 Anda ke basis data RDS Anda dengan mengonfigurasi dan menetapkan grup keamanan secara manual untuk mereproduksi konfigurasi yang secara otomatis dibuat oleh fitur koneksi otomatis di Opsi 1 dan Opsi 2.

## Konteks

Sebagai konteks mengapa Anda harus mengonfigurasi koneksi antara instans EC2 Anda dan basis data RDS, mari pertimbangkan skenario berikut: Situs web Anda menyajikan formulir kepada pengguna Anda untuk diisi. Anda perlu menangkap data formulir dalam basis data. Anda dapat meng-hosting situs web Anda pada instans EC2 yang telah dikonfigurasi sebagai server web, dan Anda dapat menangkap data formulir dalam basis data RDS. Instans EC2 dan basis data RDS harus terhubung satu sama lain sehingga data formulir dapat keluar dari instans EC2 menuju basis data

RDS. Tutorial ini menjelaskan cara mengonfigurasi koneksi itu. Perhatikan bahwa ini hanyalah salah satu contoh kasus penggunaan untuk menghubungkan instans EC2 dan basis data RDS.

## Arsitektur

Diagram berikut menunjukkan sumber daya yang dibuat dan konfigurasi arsitektur yang dihasilkan dari menyelesaikan semua langkah dalam tutorial ini.

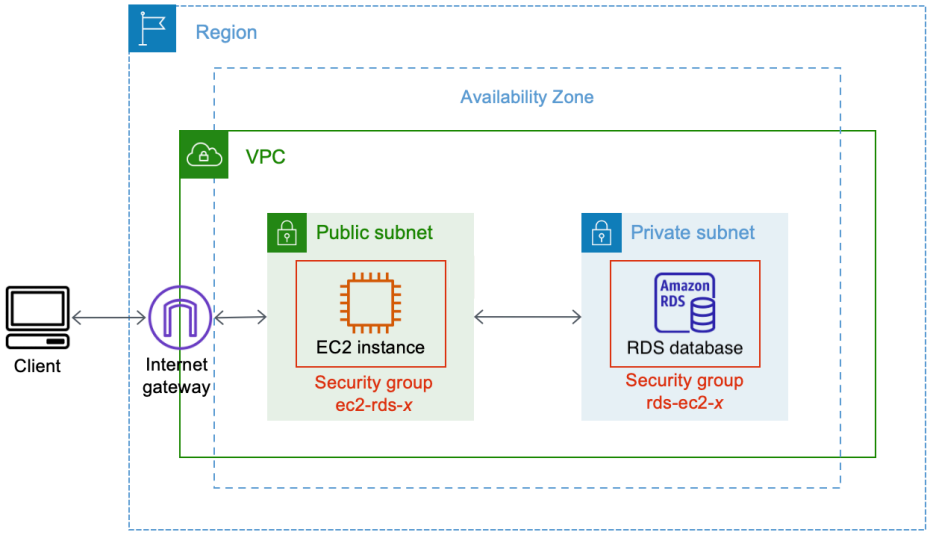



Diagram ini menggambarkan sumber daya berikut yang akan Anda buat:

- Anda akan membuat instance EC2 dan database RDS di VPC Wilayah AWS, dan Availability Zone yang sama.
- Anda akan membuat instans EC2 di subnet publik.
- Anda akan membuat basis data RDS di subnet privat.

  Saat Anda menggunakan konsol RDS untuk membuat basis data RDS dan secara otomatis menghubungkan instans EC2, VPC, grup subnet DB, dan pengaturan akses publik untuk basis data dipilih secara otomatis. Basis data RDS secara otomatis dibuat dalam subnet privat dalam VPC yang sama dengan instans EC2.
- Pengguna internet dapat terhubung ke instance EC2 dengan menggunakan SSH atau HTTP/HTTPS melalui gateway Internet.
- Pengguna internet tidak dapat terhubung langsung ke basis data RDS; hanya instans EC2 yang terhubung ke basis data RDS.
- Saat Anda menggunakan fitur koneksi otomatis untuk mengizinkan lalu lintas antara instans EC2 dan basis data RDS, grup keamanan berikut secara otomatis dibuat dan ditambahkan:

- Grup keamanan ec2-rds- ***x*** dibuat dan ditambahkan ke instance EC2. Ini memiliki satu aturan keluar yang mereferensikan grup *x* keamanan rds-ec2- sebagai tujuannya. Hal ini memungkinkan lalu lintas dari instance EC2 untuk mencapai database RDS dengan kelompok keamanan rds-ec2. ***x***
- Grup keamanan rds-ec2- ***x*** dibuat dan ditambahkan ke database RDS. Ini memiliki satu aturan masuk yang mereferensikan grup *x* keamanan ec2-rds- sebagai sumbernya. Hal ini memungkinkan lalu lintas dari instance EC2 dengan grup *x* keamanan ec2-rds-untuk mencapai database RDS.

Dengan grup keamanan terpisah (satu untuk instans EC2, dan satu untuk basis data RDS), Anda memiliki kontrol yang lebih baik atas keamanan instans dan basis data. Jika Anda menggunakan grup keamanan yang sama pada instans dan basis data, kemudian memodifikasi grup keamanan agar sesuai dengan, katakanlah, hanya basis data, modifikasi akan memengaruhi instans dan basis data. Dengan kata lain, jika Anda menggunakan satu grup keamanan, Anda dapat secara tidak sengaja memodifikasi keamanan sumber daya (baik instans atau basis data) karena Anda lupa bahwa grup keamanan telah dilampirkan padanya.

Grup keamanan yang dibuat secara otomatis juga menghormati hak akses paling rendah karena mereka hanya mengizinkan koneksi timbal balik untuk beban kerja ini pada port basis data dengan membuat pasangan grup keamanan yang spesifik beban kerja.

## Pertimbangan-pertimbangan

Pertimbangkan hal-hal berikut saat Anda menyelesaikan tugas dalam tutorial ini:

- Dua konsol – Anda akan menggunakan dua konsol berikut untuk tutorial ini:
  - Konsol Amazon EC2 — Anda akan menggunakan konsol EC2 untuk meluncurkan instans, untuk secara otomatis menghubungkan instans EC2 ke basis data RDS, dan sebagai opsi manual untuk mengonfigurasi koneksi dengan membuat grup keamanan.
  - Konsol Amazon RDS — Anda akan menggunakan konsol RDS untuk membuat basis data RDS dan menghubungkan instans EC2 secara otomatis ke basis data RDS.
- Satu VPC — Untuk menggunakan fitur koneksi otomatis, instans EC2 dan basis data RDS Anda harus berada dalam VPC yang sama.

  Jika Anda secara manual mengonfigurasi koneksi antara instans EC2 dan basis data RDS, Anda dapat meluncurkan instans EC2 dalam satu VPC dan basis data RDS Anda di VPC lain; namun,

Anda perlu menyiapkan perutean tambahan dan konfigurasi VPC. Skenario ini tidak dibahas dalam tutorial ini.

- Satu Wilayah AWS — Instans EC2 dan database RDS harus berada di Wilayah yang sama.
- Dua grup keamanan – Konektivitas antara instans EC2 dan basis data RDS dikonfigurasikan oleh dua grup keamanan—grup keamanan untuk instans EC2 Anda dan grup keamanan untuk basis data RDS.

  Saat Anda menggunakan fitur koneksi otomatis di konsol EC2 atau konsol RDS untuk mengonfigurasi konektivitas (Opsi 1 dan Opsi 2 tutorial ini), grup keamanan secara otomatis dibuat dan ditetapkan ke instans EC2 dan basis data RDS.

  Jika Anda tidak menggunakan fitur koneksi otomatis, Anda harus membuat dan menetapkan grup keamanan secara manual. Anda melakukan ini di Opsi 3 dari tutorial ini.

## Waktu untuk menyelesaikan tutorial

30 menit

Anda dapat menyelesaikan seluruh tutorial dalam sekali duduk, atau Anda dapat menyelesaikan tugas satu per satu.

## Biaya

Dengan menyelesaikan tutorial ini, Anda mungkin mengeluarkan biaya untuk AWS sumber daya yang Anda buat.

Anda dapat menggunakan Amazon EC2 di bawah Tingkat Gratis as alkan AWS akun Anda memenuhi syarat untuk Tingkat Gratis dan Anda mengonfigurasi sumber daya Anda sesuai dengan persyaratan Tingkat Gratis. Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.

Jika instans EC2 dan basis data RDS Anda berada di Zona Ketersediaan yang berbeda, Anda akan dikenai biaya transfer data. Untuk menghindari biaya ini, instans EC2 dan basis data RDS harus berada dalam Zona Ketersediaan yang sama. Untuk informasi tentang biaya transfer data, lihat Transfer Data di halaman On-Demand Harga Amazon EC2.

Untuk mencegah timbulnya biaya setelah Anda menyelesaikan tutorial, pastikan untuk menghapus sumber daya jika tidak lagi diperlukan. Untuk langkah-langkah menghapus sumber daya, lihat Tugas 4 (Opsional): Bersihkan.

# Opsi 1: Secara otomatis menghubungkan instance ke database RDS menggunakan konsol EC2

Tujuan Opsi 1 adalah mengeksplorasi fitur koneksi otomatis di konsol EC2 yang secara otomatis mengonfigurasi koneksi antara instans EC2 dan basis data RDS Anda untuk memungkinkan lalu lintas antara instans EC2 dan basis data RDS. Di Opsi 3, Anda akan mempelajari cara mengonfigurasi koneksi secara manual.

Tugas


- Sebelum Anda mulai
- Tugas 1 (Opsional): Buat database RDS
- Tugas 2 (Opsional): Luncurkan instance EC2
- Tugas 3: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda
- Tugas 4: Verifikasi konfigurasi koneksi
- Tugas 5 (Opsional): Bersihkan


## Sebelum Anda mulai

Anda memerlukan hal berikut ini untuk menyelesaikan tutorial ini:

- Basis data RDS yang berada dalam VPC yang sama dengan instans EC2. Anda dapat menggunakan basis data RDS yang ada atau mengikuti langkah-langkah di Tugas 1 untuk membuat basis data RDS baru.
- Instans EC2 yang berada dalam VPC yang sama dengan basis data RDS. Anda dapat menggunakan instans EC2 yang ada atau mengikuti langkah-langkah di Tugas 2 untuk membuat instans EC2 baru.
- Izin untuk memanggil operasi berikut ini:
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSecurityGroup`
  - `ec2:CreateSubnet`
  - `ec2:DescribeInstances`

- `ec2:DescribeNetworkInterfaces`
- `ec2:DescribeRouteTables`
- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:ModifyNetworkInterfaceAttribute`
- `ec2:RevokeSecurityGroupEgress`

## Tugas 1 (Opsional): Buat database RDS

> **Note**
>
> Membuat basis data Amazon RDS bukanlah fokus dari tutorial ini. Jika sudah memiliki basis data RDS dan ingin menggunakannya dalam tutorial ini, Anda dapat melewati tugas ini. Jika Anda menggunakan basis data RDS yang ada, pastikan basis data berada di VPC yang sama dengan instans EC2 Anda sehingga Anda dapat menggunakan fitur koneksi otomatis.

Tujuan dari tugas ini adalah untuk membuat basis data RDS sehingga Anda dapat menyelesaikan Tugas 3 di mana Anda akan mengonfigurasi koneksi antara instans EC2 dan basis data RDS Anda. Langkah-langkah dalam tugas ini mengonfigurasi basis data RDS sebagai berikut:

- Tipe mesin: MySQL
- Templat: Tingkat gratis
- Pengidentifikasi instans DB: **`tutorial-database-1`**
- Kelas instans DB: `db.t3.micro`

> **Important**
>
> Dalam lingkungan produksi, Anda harus mengonfigurasi basis data Anda untuk memenuhi kebutuhan spesifik Anda.

Untuk membuat basis data MySQL RDS

1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.

2. Dari pemilih Wilayah (di kanan atas), pilih sebuah Wilayah AWS. Basis data dan instans EC2 harus berada di Wilayah yang sama untuk menggunakan fitur koneksi otomatis di konsol EC2.
3. Pada dasbor, pilih Buat basis data.
4. Pada Pilih metode pembuatan basis data, periksa apakah Pembuatan Standar dipilih. Jika Anda memilih Easy create, pemilih VPC tidak tersedia. Anda harus memastikan bahwa database Anda berada di VPC yang sama dengan instans EC2 Anda untuk menggunakan fitur koneksi otomatis di konsol EC2.
5. Pada Opsi mesin, untuk Tipe mesin, pilih MySQL.
6. Pada Templat, pilih contoh templat untuk memenuhi kebutuhan Anda. Untuk tutorial ini, pilih Tingkat gratis untuk membuat basis data RDS tanpa biaya. Namun, perhatikan bahwa Tingkat Gratis hanya tersedia untuk akun yang memenuhi syarat untuk Tingkat Gratis. Anda dapat membaca lebih lanjut dengan memilih tautan Info di kotak Tingkat gratis.
7. Pada Pengaturan, lakukan hal berikut:

    a. Untuk Pengidentifikasi instans DB, masukkan nama untuk basis data. Untuk tutorial ini, masukkan **`tutorial-database-1`**.

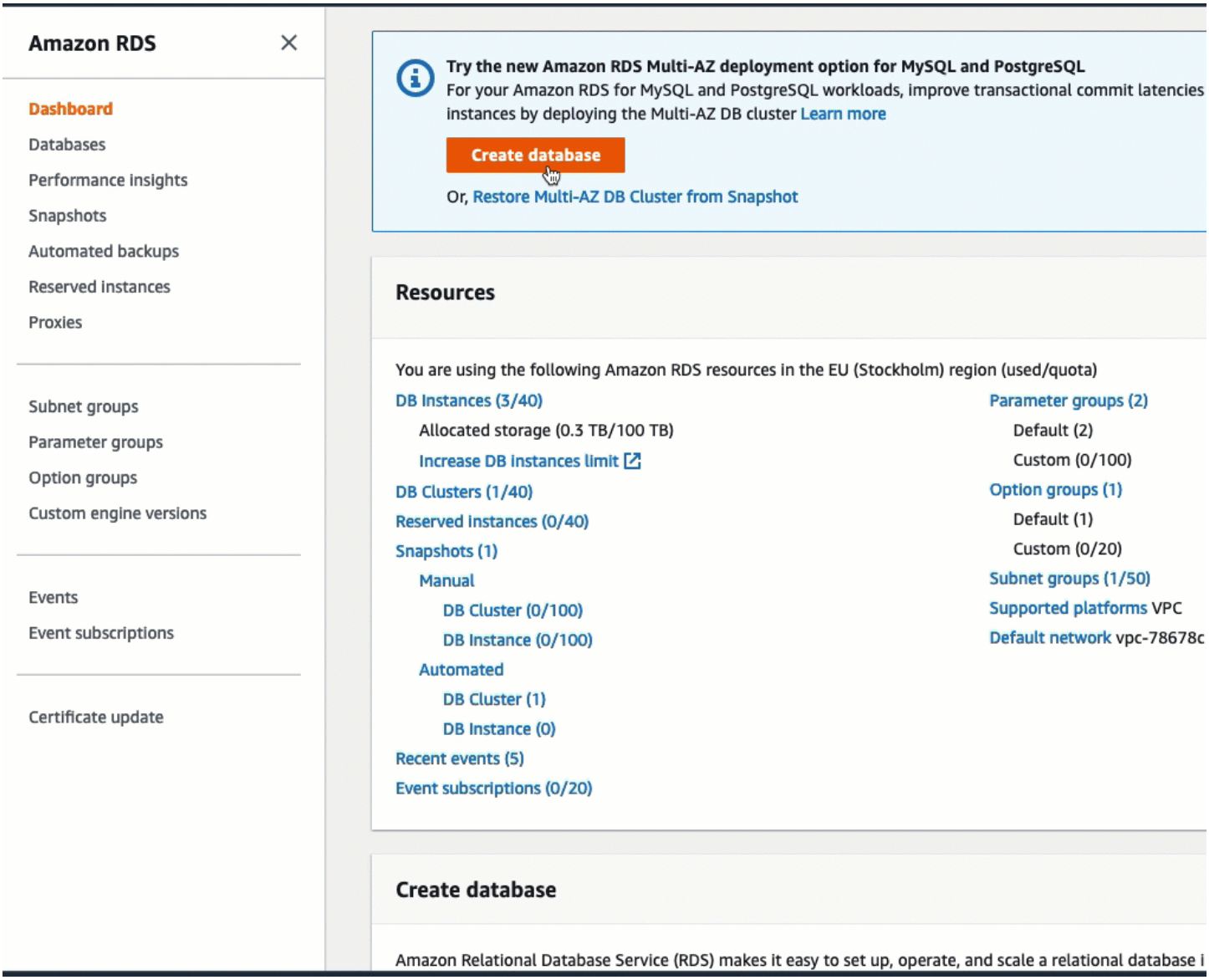
    b. Untuk Nama pengguna master, biarkan nama default, yaitu **`admin`**.

    c. Untuk Kata sandi master, masukkan kata sandi yang dapat Anda ingat untuk tutorial ini, kemudian untuk Konfirmasi kata sandi, masukkan kata sandi lagi.
8. Di bawah konfigurasi Inst ance, untuk kelas instance DB, tinggalkan default, yaitu db.t 3.micro. Jika akun Anda memenuhi syarat untuk Tingkat Gratis, Anda dapat menggunakan kelas database ini secara gratis. Untuk informasi selengkapnya, lihat AWS Tingkat Gratis.
9. Di bawah Konektivitas, untuk Sumber daya komputasi, pilih Jangan hubungkan ke sumber daya komputasi EC2 karena Anda akan menghubungkan instans EC2 dan basis data RDS nanti di Tugas 3.

    (Nantinya, di Opsi 2 tutorial ini, Anda akan mencoba fitur koneksi otomatis di konsol RDS dengan memilih Terhubung ke sumber daya komputasi EC2.)
10. Untuk cloud privat virtual (VPC), pilih VPC. VPC harus memiliki grup subnet DB. Untuk menggunakan fitur koneksi otomatis, instans EC2 dan basis data RDS harus berada di VPC yang sama.
11. Simpan semua nilai default untuk bidang lain di halaman ini.
12. Pilih Buat basis data.

Pada layar Basis Data, Status basis data baru adalah Membuat sampai basis data siap digunakan. Saat statusnya berubah menjadi Tersedia, Anda dapat terhubung ke basis data. Tergantung pada kelas basis data dan jumlah penyimpanan, diperlukan waktu hingga 20 menit sebelum basis data baru tersedia.

Lihat animasi: Membuat basis data RDS



## Tugas 2 (Opsional): Luncurkan instance EC2

> **Note**
>
> Meluncurkan sebuah instans bukanlah fokus dari tutorial ini. Jika Anda sudah memiliki instans Amazon EC2 dan ingin menggunakannya dalam tutorial ini, Anda dapat melewati tugas ini. Jika Anda menggunakan basis data RDS yang ada, pastikan basis data berada di VPC yang sama dengan basis data RDS Anda sehingga Anda dapat menggunakan fitur koneksi otomatis.

Tujuan dari tugas ini adalah untuk meluncurkan instans EC2 sehingga Anda dapat menyelesaikan Tugas 3 di mana Anda akan mengonfigurasi koneksi antara instans EC2 dan basis data Amazon RDS Anda. Langkah-langkah dalam tugas ini mengonfigurasi instans EC2 sebagai berikut:

- Nama instans: **`tutorial-instance-1`**
- AMI: Amazon Linux 2
- Tipe instans: `t2.micro`
- Auto-assign IP publik: Diaktifkan
- Grup keamanan dengan tiga aturan berikut:
  - Izinkan SSH dari alamat IP Anda
  - Izinkan lalu lintas HTTPS dari mana saja
  - Izinkan lalu lintas HTTP dari mana saja

> **Important**
>
> Dalam lingkungan produksi, Anda harus mengonfigurasi instans Anda untuk memenuhi kebutuhan spesifik Anda.

Cara meluncurkan instans EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari pemilih Wilayah (di kanan atas), pilih sebuah Wilayah AWS. Instans dan database RDS harus berada di Wilayah yang sama untuk menggunakan fitur koneksi otomatis di konsol EC2.
3. Dari Dasbor EC2, pilih Luncurkan instans.

4. Pada Nama dan tanda, masukkan nama untuk mengidentifikasi instans Anda pada Nama. Untuk tutorial ini, beri nama instans **`tutorial-instance-1`**. Meskipun nama instans tidak wajib, ketika Anda memilih instans Anda di konsol EC2, nama tersebut akan membantu Anda mengidentifikasinya dengan mudah.
5. Pada Gambar Aplikasi dan OS, pilih AMI yang memenuhi kebutuhan server web Anda. Tutorial ini menggunakan Amazon Linux 2.
6. Pada Tipe instans, pilih tipe instans yang memenuhi kebutuhan server web Anda pada Tipe instans. Tutorial ini menggunakan `t2.micro`.

> **Note**
>
> Tergantung pada saat Anda membuat akun, Anda mungkin memenuhi syarat untuk menggunakan Amazon EC2 di bawah Tingkat Gratis.
> Jika Anda membuat Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan Amazon EC2 di bawah Tingkat Gratis dengan memilih jenis instans t2.micro, atau jenis instans t 3.micro di Wilayah di mana t2.micro tidak tersedia. Ketahuilah bahwa ketika Anda meluncurkan instance t3.micro, default ke mode Unlimited, yang mungkin dikenakan biaya tambahan berdasarkan penggunaan CPU. Jika jenis instans dapat digunakan di bawah Tingkat Gratis, itu diberi label Lay ak tingkat Gratis.
> Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan jenis instans t3.micro, t3.small, t4g.micro, t4g.small, c7i-flex.large, dan m7i-flex.large selama 6 bulan atau sampai kredit Anda habis.
> Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.

7. Pada Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci Anda.
8. Pada Pengaturan jaringan, lakukan hal berikut:

   a. Untuk Jaringan dan Subnet, jika Anda belum membuat perubahan pada VPC atau subnet default Anda, Anda dapat mempertahankan pengaturan default.

      Jika Anda telah membuat perubahan pada VPC atau subnet default Anda, periksa hal berikut:

      i. Instans harus berada dalam VPC yang sama dengan basis data RDS untuk menggunakan fitur koneksi otomatis. Secara default, Anda hanya memiliki satu VPC.

ii. VPC tempat Anda meluncurkan instans harus memiliki gateway internet yang melekat padanya, sehingga Anda dapat mengakses server web dari internet. VPC default Anda secara otomatis disiapkan dengan gateway internet.

iii. Untuk memastikan instans Anda menerima alamat IP publik, untuk IP Auto-assign publik, periksa apakah Aktif kan dipilih. Jika N onaktifkan dipilih, pilih Edit (di sebelah kanan Pengaturan Jaringan), lalu, untuk IP Auto-assign publik, pilih Aktifkan.

b. Untuk terhubung ke instans Anda dengan menggunakan SSH, Anda memerlukan aturan grup keamanan yang mengotorisasi lalu lintas SSH (Linux) atau RDP (Windows) dari alamat IPv4 publik komputer Anda. Secara default, saat Anda meluncurkan sebuah instans, grup keamanan baru dibuat dengan aturan yang memungkinkan lalu lintas SSH masuk dari mana saja.

   Untuk memastikan bahwa hanya alamat IP Anda yang dapat terhubung ke instance Anda, di bawah Firewall (grup keamanan), dari daftar drop-down di samping kotak centang Iz inkan lalu lintas SSH dari, pilih IP Saya.

c. Untuk mengizinkan lalu lintas dari internet ke instans Anda, pilih kotak centang berikut:

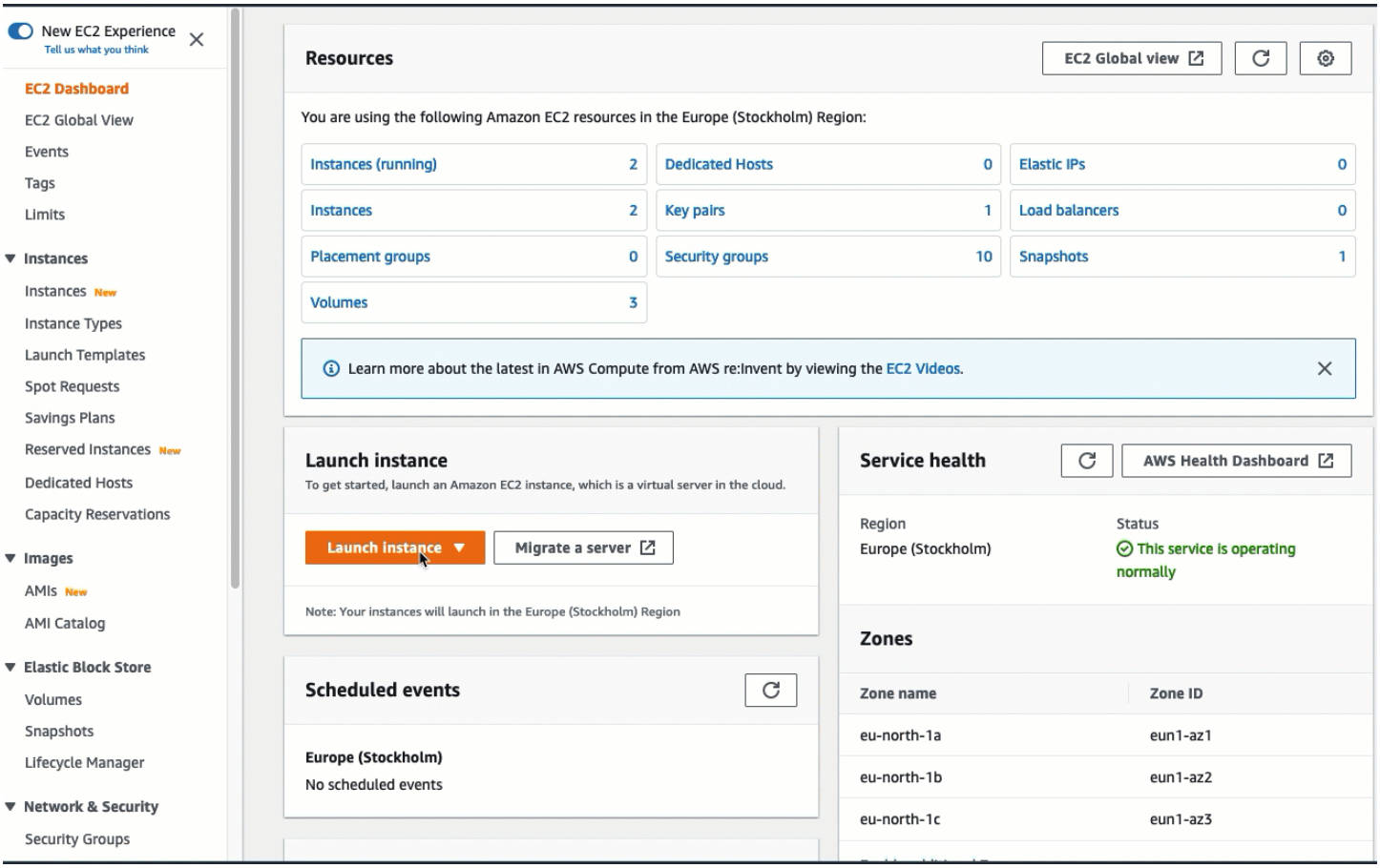
   - Izinkan lalu lintas HTTPS dari internet
   - Izinkan lalu lintas HTTP dari internet

9. Di panel Ringkasan, tinjau konfigurasi instans Anda dan pilih Luncurkan instans.
10. Biarkan halaman konfirmasi tetap terbuka. Anda akan membutuhkannya untuk tugas berikutnya saat Anda secara otomatis menghubungkan instans Anda ke basis data Anda.

    Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

Untuk informasi tentang peluncuran instans, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

Lihat animasi: Luncurkan instans EC2




## Tugas 3: Hubungkan instans EC2 Anda secara otomatis ke basis data RDS Anda

Tujuan dari tugas ini adalah untuk menggunakan fitur koneksi otomatis di konsol EC2 untuk secara otomatis mengonfigurasi koneksi antara instans EC2 Anda dan basis data RDS Anda.

Untuk secara otomatis menghubungkan instans EC2 ke basis data RDS menggunakan konsol EC2

1. Pada halaman konfirmasi peluncuran instans (harus terbuka dari tugas sebelumnya), pilih Hubungkan basis data RDS.

   Jika Anda menutup halaman konfirmasi, ikuti langkah-langkah berikut:

   a. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

   b. Di panel navigasi, pilih Instans.

   c. Pilih instans EC2 yang baru saja Anda buat, lalu pilih Actions, Jaringan, basis data Connect RDS.

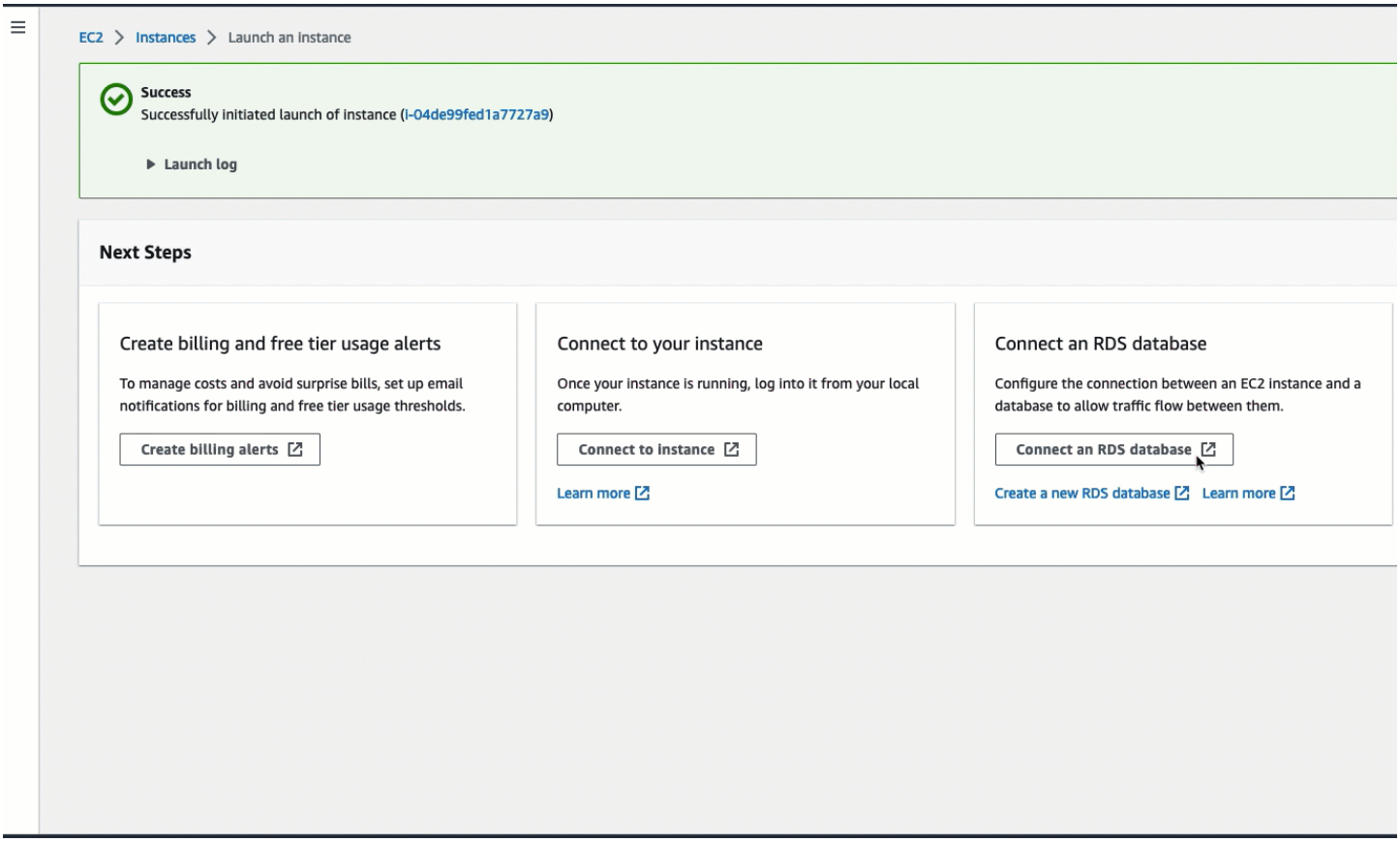
Jika Hubungkan basis data RDS tidak tersedia, periksa apakah instans EC2 berada dalam status Berjalan.

2. Untuk Peran basis data, pilih Instans. Instans dalam hal ini mengacu pada instans basis data.
3. Untuk basis data RDS, pilih basis data RDS yang Anda buat di Tugas 1.

> **Note**
>
> Instans EC2 dan database RDS harus berada di VPC yang sama untuk terhubung satu sama lain.

4. Pilih Hubungkan.

Lihat animasi: Menghubungkan instans EC2 yang baru diluncurkan ke basis data RDS




## Tugas 4: Verifikasi konfigurasi koneksi

Tujuan dari tugas ini adalah untuk memverifikasi bahwa dua kelompok keamanan dibuat dan ditetapkan ke instans dan basis data.

Saat Anda menggunakan fitur koneksi otomatis di konsol untuk mengonfigurasi konektivitas, grup keamanan secara otomatis dibuat dan ditetapkan ke instance dan database, sebagai berikut:

- Grup keamanan rds-ec2- ***x*** dibuat dan ditambahkan ke database RDS. Ini memiliki satu aturan masuk yang mereferensikan grup *x* keamanan ec2-rds- sebagai sumbernya. Hal ini memungkinkan lalu lintas dari instance EC2 dengan grup *x* keamanan ec2-rds-untuk mencapai database RDS.
- Grup keamanan ec2-rds- ***x*** dibuat dan ditambahkan ke instance EC2. Ini memiliki satu aturan keluar yang mereferensikan grup *x* keamanan rds-ec2- sebagai tujuannya. Hal ini memungkinkan lalu lintas dari instance EC2 untuk mencapai database RDS dengan kelompok keamanan rds-ec2. ***x***

Untuk memverifikasi konfigurasi koneksi menggunakan konsol

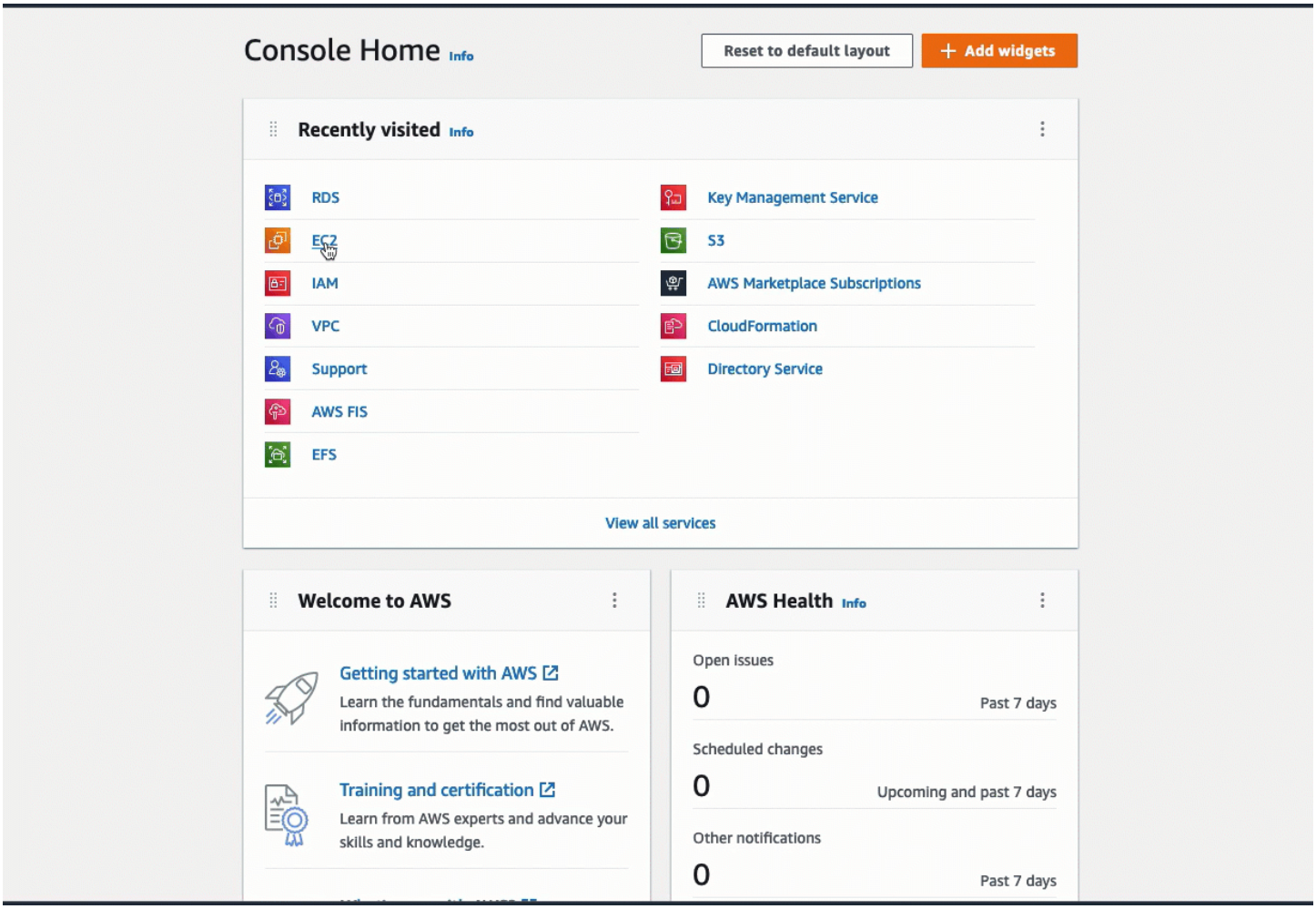
1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.
2. Di halaman navigasi, pilih Basis Data.
3. Pilih basis data RDS yang Anda buat untuk tutorial ini.
4. Pada tab Konektivitas & keamanan, di bawah Keamanan, grup keamanan VPC, verifikasi bahwa grup keamanan bernama rds-ec2- ditampilkan. ***x***
5. Pilih grup keamanan rds-ec2- ***x***. Layar Grup Keamanan di konsol EC2 terbuka.
6. Pilih grup *x* keamanan rds-ec2- untuk membukanya.
7. Pilih tab Aturan masuk.
8. Verifikasi bahwa aturan grup keamanan berikut ada, sebagai berikut:
   - Tipe: MYSQL/Aurora
   - Rentang port: 3306
   - Sumber: ***sg-0987654321example***/ec2-rds- ***x*** — Ini adalah grup keamanan yang ditetapkan ke instans EC2 yang Anda verifikasi pada langkah sebelumnya.
   - Deskripsi: A turan untuk mengizinkan koneksi dari instans EC2 dengan terlampir ***sg-1234567890example***
9. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
10. Di panel navigasi, pilih Instans.
11. Pilih instans EC2 yang Anda pilih untuk terhubung ke basis data RDS di tugas sebelumnya, dan pilih tab Keamanan.

12. Di bawah Detail keamanan, Grup keamanan, verifikasi bahwa grup keamanan yang disebut ec2-rds- ***x*** ada dalam daftar. *x* adalah angka.
13. Pilih grup *x* keamanan ec2-rds- untuk membukanya.
14. Pilih tab Aturan keluar.
15. Verifikasi bahwa aturan grup keamanan berikut ada, sebagai berikut:

    - Tipe: MYSQL/Aurora
    - Rentang port: 3306
    - Tujuan: ***sg-1234567890example***/rds-ec2- ***x***
    - Deskripsi: Aturan untuk mengizinkan koneksi ke **`database-tutorial`** dari setiap instans grup keamanan ini dilampirkan ke

Dengan memverifikasi bahwa grup keamanan dan aturan grup keamanan ini ada dan bahwa masing-masing ditetapkan ke basis data RDS dan instans EC2 seperti yang dijelaskan dalam prosedur ini, Anda dapat memverifikasi bahwa koneksi secara otomatis dikonfigurasi dengan menggunakan fitur koneksi otomatis.

Lihat animasi: Verifikasi konfigurasi koneksi




Anda telah menyelesaikan Opsi 1 dari tutorial ini. Anda sekarang dapat menyelesaikan Opsi 2, yang mengajarkan Anda cara menggunakan konsol RDS untuk secara otomatis menghubungkan instans EC2 ke basis data RDS, atau Anda dapat menyelesaikan Opsi 3, yang mengajarkan Anda cara mengonfigurasi secara manual grup keamanan yang dibuat di Opsi 1 secara otomatis.

## Tugas 5 (Opsional): Bersihkan

**Warning**

Mengakhiri instance bersifat permanen dan tidak dapat diubah.
Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.

Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Sekarang setelah Anda menyelesaikan tutorial, itu adalah praktik yang baik untuk membersihkan (menghapus) sumber daya apa pun yang tidak ingin Anda gunakan lagi. Membersihkan AWS sumber daya mencegah akun Anda menimbulkan biaya lebih lanjut.

Jika Anda membuat basis data RDS khusus untuk tutorial ini, Anda dapat mengakhirinya untuk menghentikan biaya apa pun yang terkait dengannya.

Untuk mengakhiri instans menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans yang Anda buat untuk tutorial ini, dan pilih Status instans, Akhiri instans.
4. Pilih Akhiri saat diminta untuk konfirmasi.

Jika Anda membuat basis data RDS khusus untuk tutorial ini, Anda dapat menghapusnya untuk menghentikan biaya apa pun yang terkait dengannya.

Untuk menghapus basis data RDS menggunakan konsol

1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.
2. Di panel navigasi, pilih Basis Data.
3. Pilih basis data RDS yang Anda buat untuk tutorial ini, dan pilih Tindakan, Hapus.
4. Masukkan **`delete me`** di dalam kotak, lalu pilih Hapus.

## Opsi 2: Secara otomatis menghubungkan instance ke database RDS menggunakan konsol RDS

Tujuan Opsi 2 adalah mengeksplorasi fitur koneksi otomatis di konsol RDS yang secara otomatis mengonfigurasi koneksi antara instans EC2 dan basis data RDS Anda untuk memungkinkan lalu lintas antara instans EC2 dan basis data RDS. Di Opsi 3, Anda akan mempelajari cara mengonfigurasi koneksi secara manual.

Tugas

- Sebelum Anda mulai
- Tugas 1 (Opsional): Luncurkan instance EC2
- Tugas 2: Buat basis data RDS dan secara otomatis hubungkan ke instans EC2 Anda
- Tugas 3: Verifikasi konfigurasi koneksi
- Tugas 4 (Opsional): Bersihkan


## Sebelum Anda mulai

Anda memerlukan hal berikut ini untuk menyelesaikan tutorial ini:

- Instans EC2 yang berada dalam VPC yang sama dengan basis data RDS. Anda dapat menggunakan instans EC2 yang ada atau mengikuti langkah-langkah di Tugas 1 untuk membuat instans baru.
- Izin untuk memanggil operasi berikut ini:
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSecurityGroup`
  - `ec2:CreateSubnet`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeSubnets`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

## Tugas 1 (Opsional): Luncurkan instance EC2

> **Note**
>
> Meluncurkan sebuah instans bukanlah fokus dari tutorial ini. Jika Anda sudah memiliki instans Amazon EC2 dan ingin menggunakannya dalam tutorial ini, Anda dapat melewati tugas ini.

Tujuan dari tugas ini adalah untuk meluncurkan instans EC2 sehingga Anda dapat menyelesaikan Tugas 2 di mana Anda akan mengonfigurasi koneksi antara instans EC2 dan basis data Amazon RDS Anda. Langkah-langkah dalam tugas ini mengonfigurasi instans EC2 sebagai berikut:

- Nama instans: **`tutorial-instance-2`**
- AMI: Amazon Linux 2
- Tipe instans: `t2.micro`
- Auto-assign IP publik: Diaktifkan
- Grup keamanan dengan tiga aturan berikut:
  - Izinkan SSH dari alamat IP Anda
  - Izinkan lalu lintas HTTPS dari mana saja
  - Izinkan lalu lintas HTTP dari mana saja

> **Important**
>
> Dalam lingkungan produksi, Anda harus mengonfigurasi instans Anda untuk memenuhi kebutuhan spesifik Anda.

Cara meluncurkan instans EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari Dasbor EC2, pilih Luncurkan instans.
3. Pada Nama dan tanda, masukkan nama untuk mengidentifikasi instans Anda pada Nama. Untuk tutorial ini, beri nama instans **`tutorial-instance-2`**. Meskipun nama instans tidak wajib, ketika Anda memilih instans Anda di konsol RDS, nama tersebut akan membantu Anda mengidentifikasinya dengan mudah.
4. Pada Gambar Aplikasi dan OS, pilih AMI yang memenuhi kebutuhan server web Anda. Tutorial ini menggunakan Amazon Linux.
5. Pada Tipe instans, pilih tipe instans yang memenuhi kebutuhan server web Anda pada Tipe instans. Tutorial ini menggunakan `t2.micro`.

> **Note**
>
> Tergantung pada saat Anda membuat akun, Anda mungkin memenuhi syarat untuk menggunakan Amazon EC2 di bawah Tingkat Gratis.
> Jika Anda membuat Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan Amazon EC2 di bawah Tingkat Gratis dengan memilih jenis instans t2.micro, atau jenis instans t 3.micro di Wilayah di mana t2.micro tidak tersedia. Ketahuilah bahwa ketika Anda meluncurkan instance t3.micro, default ke mode Unlimited, yang mungkin dikenakan biaya tambahan berdasarkan penggunaan CPU. Jika jenis instans dapat digunakan di bawah Tingkat Gratis, itu diberi label Lay ak tingkat Gratis.
> Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan jenis instans t3.micro, t3.small, t4g.micro, t4g.small, c7i-flex.large, dan m7i-flex.large selama 6 bulan atau sampai kredit Anda habis.
> Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.

6. Pada Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci Anda.
7. Pada Pengaturan jaringan, lakukan hal berikut:

   a. Untuk Jaringan dan Subnet, jika Anda belum membuat perubahan pada VPC atau subnet default Anda, Anda dapat mempertahankan pengaturan default.

      Jika Anda telah membuat perubahan pada VPC atau subnet default Anda, periksa hal berikut:

      i. Instans harus berada dalam VPC yang sama dengan basis data RDS untuk menggunakan konfigurasi koneksi otomatis. Secara default, Anda hanya memiliki satu VPC.

      ii. VPC tempat Anda meluncurkan instans harus memiliki gateway internet yang melekat padanya, sehingga Anda dapat mengakses server web dari internet. VPC default Anda secara otomatis disiapkan dengan gateway internet.

      iii. Untuk memastikan instans Anda menerima alamat IP publik, untuk IP Auto-assign publik, periksa apakah Aktif kan dipilih. Jika N onaktifkan dipilih, pilih Edit (di sebelah kanan Pengaturan Jaringan), lalu, untuk IP Auto-assign publik, pilih Aktifkan.

b. Untuk terhubung ke instans Anda dengan menggunakan SSH, Anda memerlukan aturan grup keamanan yang mengotorisasi lalu lintas SSH (Linux) atau RDP (Windows) dari alamat IPv4 publik komputer Anda. Secara default, saat Anda meluncurkan sebuah instans, grup keamanan baru dibuat dengan aturan yang memungkinkan lalu lintas SSH masuk dari mana saja.

   Untuk memastikan bahwa hanya alamat IP Anda yang dapat terhubung ke instance Anda, di bawah  Firewall (grup keamanan), dari daftar drop-down di samping  kotak centang Iz  inkan lalu lintas SSH dari, pilih IP  Saya.

c. Untuk mengizinkan lalu lintas dari internet ke instans Anda, pilih kotak centang berikut:

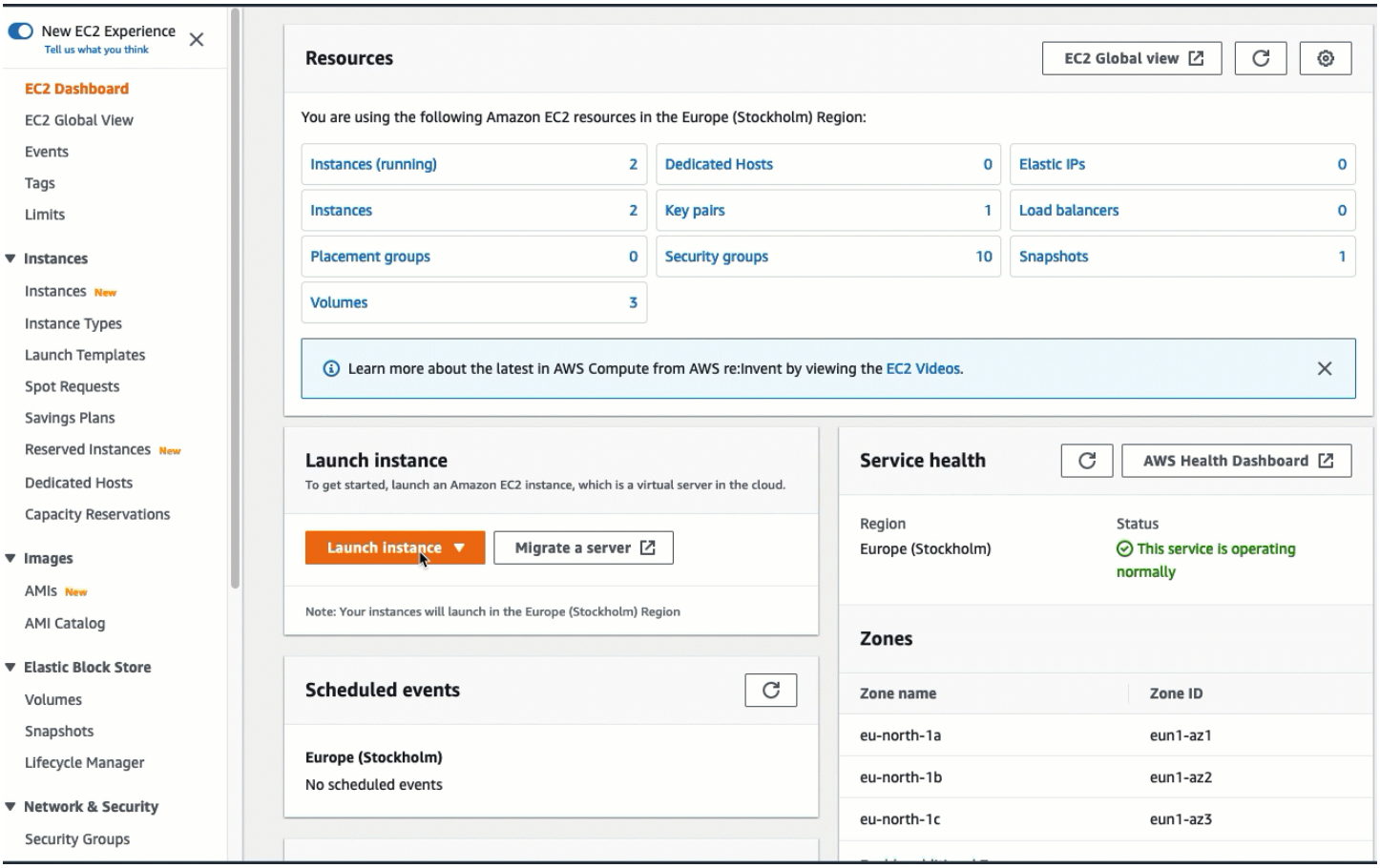
   - Izinkan lalu lintas HTTPS dari internet
   - Izinkan lalu lintas HTTP dari internet

8. Di panel Ringkasan, tinjau konfigurasi instans Anda dan pilih Luncurkan instans.
9. Pilih Lihat semua instans untuk menutup halaman konfirmasi dan kembali ke konsol. Instans Anda pertama-tama akan berada dalam status `pending`, kemudian akan masuk ke status `running`.

   Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

Untuk informasi tentang peluncuran instans, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

Lihat animasi: Luncurkan instans EC2




## Tugas 2: Buat basis data RDS dan secara otomatis hubungkan ke instans EC2 Anda

Tujuan dari tugas ini adalah untuk membuat basis data RDS dan menggunakan fitur koneksi otomatis di konsol EC2 untuk secara otomatis mengonfigurasi koneksi antara instans EC2 Anda dan basis data RDS Anda. Langkah-langkah dalam tugas ini mengonfigurasi instans DB sebagai berikut:

- Tipe mesin: MySQL
- Templat: Tingkat gratis
- Pengidentifikasi instans DB: **`tutorial-database`**
- Kelas instans DB: `db.t3.micro`

> **Important**
>
> Dalam lingkungan produksi, Anda harus mengonfigurasi instans Anda untuk memenuhi kebutuhan spesifik Anda.

Untuk membuat basis data RDS dan secara otomatis menghubungkannya ke instans EC2

1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.
2. Dari pemilih Wilayah (di kanan atas), pilih Wilayah AWS tempat Anda membuat instance EC2. Instans EC2 dan basis data RDS harus berada di Wilayah yang sama.
3. Pada dasbor, pilih Buat basis data.
4. Pada Pilih metode pembuatan basis data, periksa apakah Pembuatan Standar dipilih. Jika Anda memilih Mudah buat, fitur koneksi otomatis tidak tersedia.
5. Pada Opsi mesin, untuk Tipe mesin, pilih MySQL.
6. Pada Templat, pilih contoh templat untuk memenuhi kebutuhan Anda. Untuk tutorial ini, pilih Tingkat gratis untuk membuat basis data RDS tanpa biaya. Namun, perhatikan bahwa Tingkat Gratis hanya tersedia jika akun Anda memenuhi syarat untuk Tingkat Gratis. Anda dapat membaca lebih lanjut dengan memilih tautan Info di kotak Tingkat gratis.
7. Pada Pengaturan, lakukan hal berikut:
   a. Untuk Pengidentifikasi instans DB, masukkan nama untuk basis data. Untuk tutorial ini, masukkan **`tutorial-database`**.
   b. Untuk Nama pengguna master, biarkan nama default, yaitu **`admin`**.
   c. Untuk Kata sandi master, masukkan kata sandi yang dapat Anda ingat untuk tutorial ini, kemudian untuk Konfirmasi kata sandi, masukkan kata sandi lagi.
8. Di bawah konfigurasi Inst ance, untuk kelas instance DB, tinggalkan default, yaitu db.t 3.micro. Jika akun Anda memenuhi syarat untuk Tingkat Gratis, Anda dapat menggunakan instance ini secara gratis. Untuk informasi selengkapnya, lihat AWS Tingkat Gratis.
9. Untuk Konektivitas, Sumber daya komputasi, pilih Hubungkan ke sumber daya komputasi EC2. Ini adalah fitur koneksi otomatis di konsol RDS.
10. Untuk instans EC2, pilih instans EC2 yang ingin Anda connect. Untuk keperluan tutorial ini, Anda dapat memilih instans yang Anda buat di tugas sebelumnya, yang Anda beri nama**`tutorial-instance`**, atau memilih instans lain yang ada. Jika Anda tidak melihat instans Anda dalam daftar, pilih ikon refresh di sebelah kanan Konektivitas.

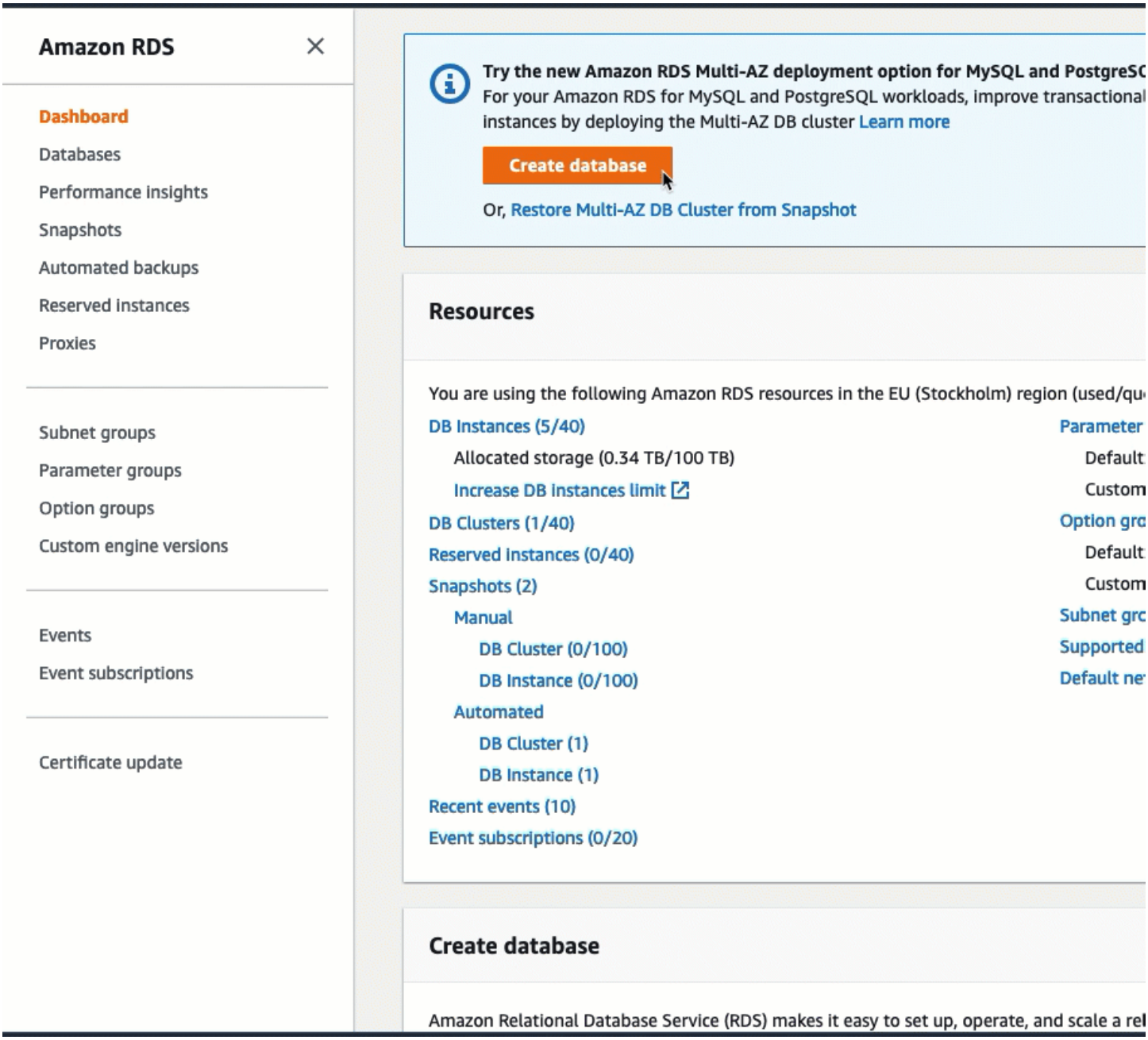
Saat Anda menggunakan fitur koneksi otomatis, grup keamanan ditambahkan ke instans EC2 ini, dan grup keamanan lain ditambahkan ke basis data RDS. Grup keamanan secara otomatis dikonfigurasi untuk memungkinkan lalu lintas antara instans EC2 dan basis data RDS. Pada tugas berikutnya, Anda akan memverifikasi bahwa grup keamanan telah dibuat dan ditetapkan ke instans EC2 dan basis data RDS.

11. Pilih Buat basis data.

    Pada layar Basis Data, Status basis data baru adalah Membuat sampai basis data siap digunakan. Saat statusnya berubah menjadi Tersedia, Anda dapat terhubung ke basis data. Tergantung pada kelas basis data dan jumlah penyimpanan, diperlukan waktu hingga 20 menit sebelum basis data baru tersedia.

Untuk mempelajari selengkapnya, lihat Mengonfigurasi konektivitas jaringan otomatis dengan instans EC2 di Panduan Pengguna Amazon RDS.

Lihat animasi: Membuat basis data RDS dan secara otomatis menghubungkannya ke instans EC2.




## Tugas 3: Verifikasi konfigurasi koneksi

Tujuan dari tugas ini adalah untuk memverifikasi bahwa dua kelompok keamanan dibuat dan ditetapkan ke instans dan basis data.

Saat Anda menggunakan fitur koneksi otomatis di konsol untuk mengonfigurasi konektivitas, grup keamanan secara otomatis dibuat dan ditetapkan ke instance dan database, sebagai berikut:

- Grup keamanan rds-ec2- ***x*** dibuat dan ditambahkan ke database RDS. Ini memiliki satu aturan masuk yang mereferensikan grup *x* keamanan ec2-rds- sebagai sumbernya. Hal ini memungkinkan lalu lintas dari instance EC2 dengan grup *x* keamanan ec2-rds-untuk mencapai database RDS.
- Grup keamanan ec2-rds- ***x*** dibuat dan ditambahkan ke instance EC2. Ini memiliki satu aturan keluar yang mereferensikan grup *x* keamanan rds-ec2- sebagai tujuannya. Hal ini memungkinkan lalu lintas dari instance EC2 untuk mencapai database RDS dengan kelompok keamanan rds-ec2. ***x***

Untuk memverifikasi konfigurasi koneksi menggunakan konsol

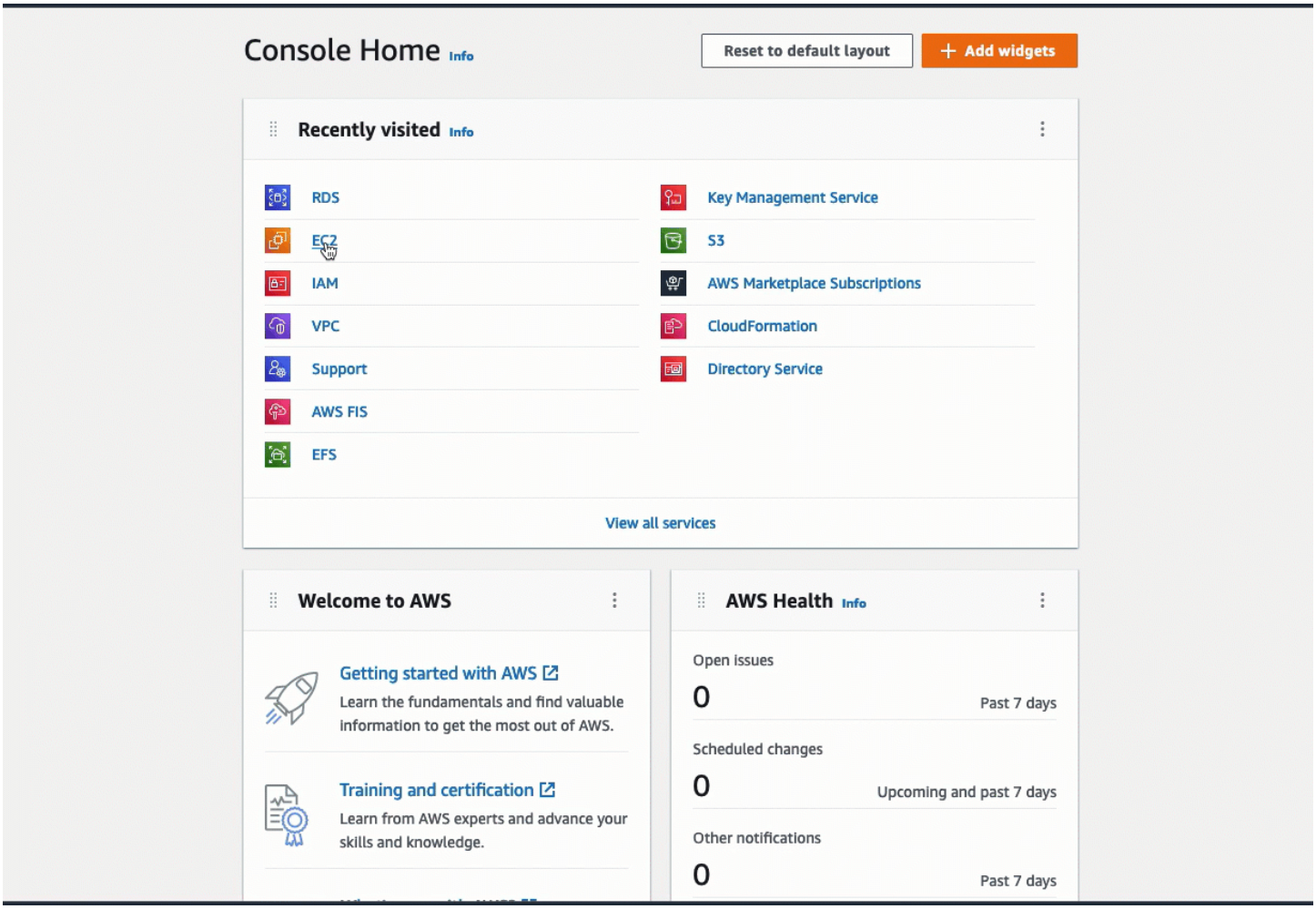
1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.
2. Di halaman navigasi, pilih Basis Data.
3. Pilih basis data RDS yang Anda buat untuk tutorial ini.
4. Pada tab Konektivitas & keamanan, di bawah Keamanan, grup keamanan VPC, verifikasi bahwa grup keamanan bernama rds-ec2- ditampilkan. ***x***
5. Pilih grup keamanan rds-ec2- ***x***. Layar Grup Keamanan di konsol EC2 terbuka.
6. Pilih grup *x* keamanan rds-ec2- untuk membukanya.
7. Pilih tab Aturan masuk.
8. Verifikasi bahwa aturan grup keamanan berikut ada, sebagai berikut:
   - Tipe: MYSQL/Aurora
   - Rentang port: 3306
   - Sumber: ***sg-0987654321example***/ec2-rds- ***x*** — Ini adalah grup keamanan yang ditetapkan ke instans EC2 yang Anda verifikasi pada langkah sebelumnya.
   - Deskripsi: A turan untuk mengizinkan koneksi dari instans EC2 dengan terlampir ***sg-1234567890example***
9. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
10. Di panel navigasi, pilih Instans.
11. Pilih instans EC2 yang Anda pilih untuk terhubung ke basis data RDS di tugas sebelumnya, dan pilih tab Keamanan.
12. Di bawah Detail keamanan, Grup keamanan, verifikasi bahwa grup keamanan yang disebut ec2-rds- ***x*** ada dalam daftar. *x*adalah angka.

13. Pilih grup *x* keamanan ec2-rds- untuk membukanya.
14. Pilih tab Aturan keluar.
15. Verifikasi bahwa aturan grup keamanan berikut ada, sebagai berikut:

    - Tipe: MYSQL/Aurora
    - Rentang port: 3306
    - Tujuan: ***sg-1234567890example***/rds-ec2- ***x***
    - Deskripsi: Aturan untuk mengizinkan koneksi ke **`database-tutorial`** dari setiap instans grup keamanan ini dilampirkan ke

Dengan memverifikasi bahwa grup keamanan dan aturan grup keamanan ini ada dan bahwa masing-masing ditetapkan ke basis data RDS dan instans EC2 seperti yang dijelaskan dalam prosedur ini, Anda dapat memverifikasi bahwa koneksi secara otomatis dikonfigurasi dengan menggunakan fitur koneksi otomatis.

Lihat animasi: Verifikasi konfigurasi koneksi




Anda telah menyelesaikan Opsi 2 dari tutorial ini. Anda sekarang dapat menyelesaikan Opsi 3, yang mengajarkan Anda cara untuk mengonfigurasi secara manual grup keamanan yang dibuat di Opsi 2 secara otomatis.

## Tugas 4 (Opsional): Bersihkan

**Warning**

Mengakhiri instance bersifat permanen dan tidak dapat diubah.
Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.

Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Sekarang setelah Anda menyelesaikan tutorial, itu adalah praktik yang baik untuk membersihkan (menghapus) sumber daya apa pun yang tidak ingin Anda gunakan lagi. Membersihkan AWS sumber daya mencegah akun Anda menimbulkan biaya lebih lanjut.

Jika Anda membuat basis data RDS khusus untuk tutorial ini, Anda dapat mengakhirinya untuk menghentikan biaya apa pun yang terkait dengannya.

Untuk mengakhiri instans menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans yang Anda buat untuk tutorial ini, dan pilih Status instans, Akhiri instans.
4. Pilih Akhiri saat diminta untuk konfirmasi.

Jika Anda membuat basis data RDS khusus untuk tutorial ini, Anda dapat menghapusnya untuk menghentikan biaya apa pun yang terkait dengannya.

Untuk menghapus basis data RDS menggunakan konsol

1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.
2. Di panel navigasi, pilih Basis Data.
3. Pilih basis data RDS yang Anda buat untuk tutorial ini, dan pilih Tindakan, Hapus.
4. Masukkan **`delete me`** di dalam kotak, lalu pilih Hapus.

## Opsi 3: Menghubungkan instance secara manual ke database RDS dengan membuat grup keamanan

Tujuan dari Opsi 3 adalah untuk mempelajari cara mengonfigurasi koneksi secara manual antara instans EC2 dan basis data RDS dengan mereproduksi konfigurasi fitur koneksi otomatis secara manual.

Tugas

- Sebelum Anda mulai
- Tugas 1 (Opsional): Luncurkan instance EC2
- Tugas 2 (Opsional): Buat database RDS
- Tugas 3: Hubungkan instans EC2 Anda secara manual ke basis data RDS Anda dengan membuat grup keamanan dan menetapkannya ke instans
- Tugas 4 (Opsional): Bersihkan


## Sebelum Anda mulai

Anda memerlukan hal berikut ini untuk menyelesaikan tutorial ini:

- Instans EC2 yang berada dalam VPC yang sama dengan basis data RDS. Anda dapat menggunakan instans EC2 yang ada atau mengikuti langkah-langkah di Tugas 1 untuk membuat instans baru.
- Basis data RDS yang berada dalam VPC yang sama dengan instans EC2. Anda dapat menggunakan basis data RDS yang ada atau mengikuti langkah-langkah di Tugas 2 untuk membuat basis data baru.
- Izin untuk memanggil operasi berikut ini:
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSecurityGroup`
  - `ec2:CreateSubnet`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`
  - `ec2:DescribeSecurityGroups`
  - `ec2:DescribeSubnets`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

## Tugas 1 (Opsional): Luncurkan instance EC2

> **Note**
>
> Meluncurkan sebuah instans bukanlah fokus dari tutorial ini. Jika Anda sudah memiliki instans Amazon EC2 dan ingin menggunakannya dalam tutorial ini, Anda dapat melewati tugas ini.

Tujuan dari tugas ini adalah untuk meluncurkan instans EC2 sehingga Anda dapat menyelesaikan Tugas 3 di mana Anda akan mengonfigurasi koneksi antara instans EC2 dan basis data Amazon RDS Anda. Langkah-langkah dalam tugas ini mengonfigurasi instans EC2 sebagai berikut:

- Nama instans: **`tutorial-instance`**
- AMI: Amazon Linux 2
- Tipe instans: `t2.micro`
- Auto-assign IP publik: Diaktifkan
- Grup keamanan dengan tiga aturan berikut:
  - Izinkan SSH dari alamat IP Anda
  - Izinkan lalu lintas HTTPS dari mana saja
  - Izinkan lalu lintas HTTP dari mana saja

> **Important**
>
> Dalam lingkungan produksi, Anda harus mengonfigurasi instans Anda untuk memenuhi kebutuhan spesifik Anda.

Cara meluncurkan instans EC2

1. Masuk ke Konsol Manajemen AWS dan buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari Dasbor EC2, pilih Luncurkan instans.
3. Pada Nama dan tanda, masukkan nama untuk mengidentifikasi instans Anda pada Nama. Untuk tutorial ini, beri nama instans **`tutorial-instance-manual-1`**. Meskipun nama instans tidak wajib, nama tersebut akan membantu Anda mengidentifikasinya dengan mudah.

4. Pada Gambar Aplikasi dan OS, pilih AMI yang memenuhi kebutuhan server web Anda. Tutorial ini menggunakan Amazon Linux.
5. Pada Tipe instans, pilih tipe instans yang memenuhi kebutuhan server web Anda pada Tipe instans. Tutorial ini menggunakan `t2.micro`.

> **Note**
>
> Tergantung pada saat Anda membuat akun, Anda mungkin memenuhi syarat untuk menggunakan Amazon EC2 di bawah Tingkat Gratis.
> Jika Anda membuat Akun AWS sebelum 15 Juli 2025 dan usianya kurang dari 12 bulan, Anda dapat menggunakan Amazon EC2 di bawah Tingkat Gratis dengan memilih jenis instans t2.micro, atau jenis instans t 3.micro di Wilayah di mana t2.micro tidak tersedia. Ketahuilah bahwa ketika Anda meluncurkan instance t3.micro, default ke mode Unlimited, yang mungkin dikenakan biaya tambahan berdasarkan penggunaan CPU. Jika jenis instans dapat digunakan di bawah Tingkat Gratis, itu diberi label Lay ak tingkat Gratis.
> Jika Anda membuat Akun AWS pada atau setelah 15 Juli 2025, Anda dapat menggunakan jenis instans t3.micro, t3.small, t4g.micro, t4g.small, c7i-flex.large, dan m7i-flex.large selama 6 bulan atau sampai kredit Anda habis.
> Untuk informasi selengkapnya, lihat Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025.

6. Pada Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci Anda.
7. Pada Pengaturan jaringan, lakukan hal berikut:

   a. Untuk Jaringan dan Subnet, jika Anda belum membuat perubahan pada VPC atau subnet default Anda, Anda dapat mempertahankan pengaturan default.

      Jika Anda telah membuat perubahan pada VPC atau subnet default Anda, periksa hal berikut:

      i. Instans harus berada dalam VPC yang sama dengan basis data RDS. Secara default, Anda hanya memiliki satu VPC.

      ii. VPC tempat Anda meluncurkan instans harus memiliki gateway internet yang melekat padanya, sehingga Anda dapat mengakses server web dari internet. VPC default Anda secara otomatis disiapkan dengan gateway internet.

iii. Untuk memastikan instans Anda menerima alamat IP publik, untuk IP Auto-assign publik, periksa apakah Aktif kan dipilih. Jika N onaktifkan dipilih, pilih Edit (di sebelah kanan Pengaturan Jaringan), lalu, untuk IP Auto-assign publik, pilih Aktifkan.

b. Untuk terhubung ke instans Anda dengan menggunakan SSH, Anda memerlukan aturan grup keamanan yang mengotorisasi lalu lintas SSH (Linux) atau RDP (Windows) dari alamat IPv4 publik komputer Anda. Secara default, saat Anda meluncurkan sebuah instans, grup keamanan baru dibuat dengan aturan yang memungkinkan lalu lintas SSH masuk dari mana saja.

   Untuk memastikan bahwa hanya alamat IP Anda yang dapat terhubung ke instance Anda, di bawah Firewall (grup keamanan), dari daftar drop-down di samping kotak centang Iz inkan lalu lintas SSH dari, pilih IP Saya.

c. Untuk mengizinkan lalu lintas dari internet ke instans Anda, pilih kotak centang berikut:

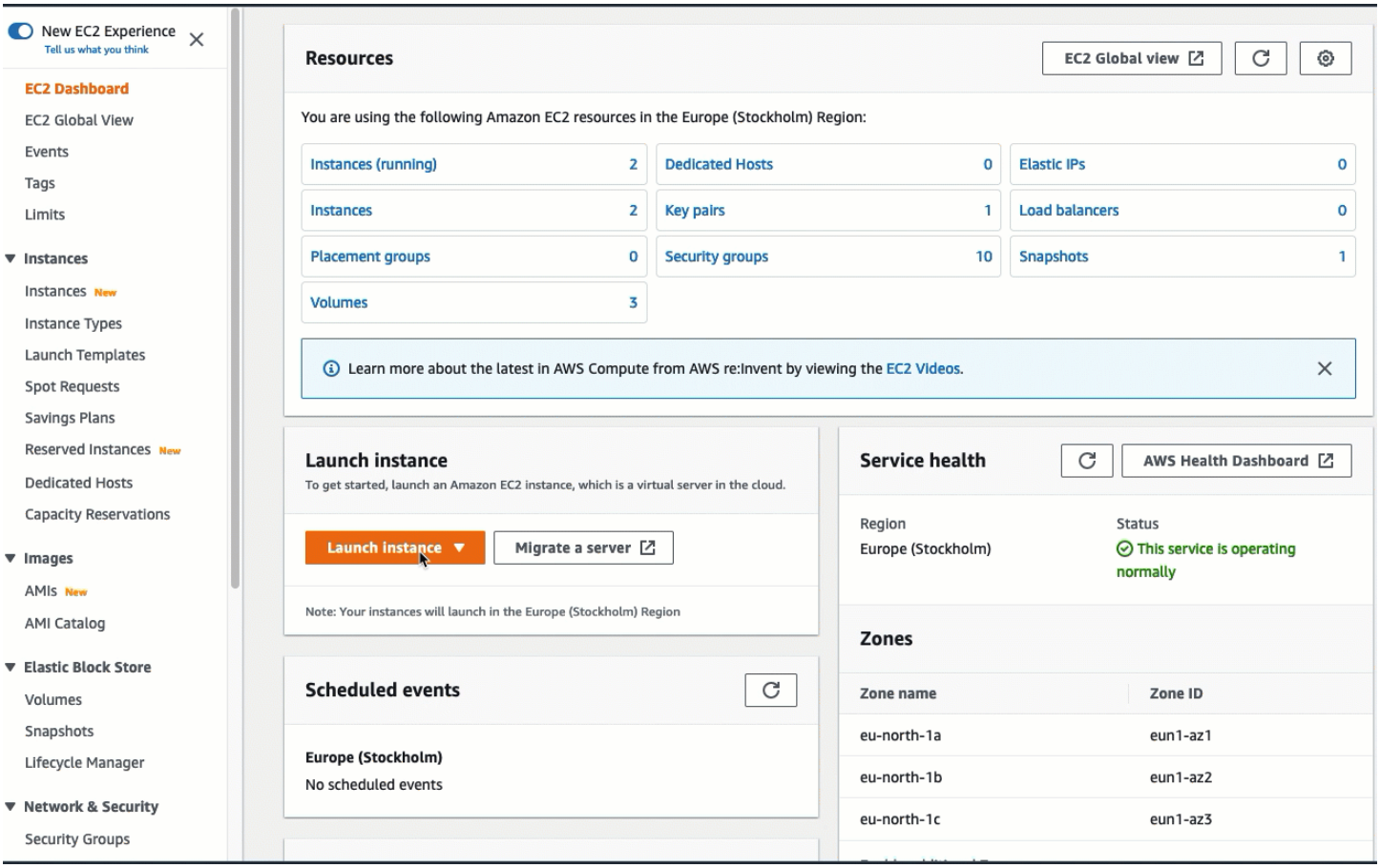
   - Izinkan lalu lintas HTTPS dari internet
   - Izinkan lalu lintas HTTP dari internet

8. Di panel Ringkasan, tinjau konfigurasi instans Anda dan pilih Luncurkan instans.
9. Pilih Lihat semua instans untuk menutup halaman konfirmasi dan kembali ke konsol. Instans Anda pertama-tama akan berada dalam status `pending`, kemudian akan masuk ke status `running`.

   Jika instans gagal diluncurkan atau status langsung menjadi `terminated`, bukan `running`, lihat Memecahkan masalah peluncuran instans Amazon EC2.

Untuk informasi tentang peluncuran instans, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

Lihat animasi: Luncurkan instans EC2




## Tugas 2 (Opsional): Buat database RDS

> **Note**
>
> Membuat basis data Amazon RDS bukanlah fokus dari tutorial ini. Jika sudah memiliki basis data RDS dan ingin menggunakannya untuk tutorial ini, Anda dapat melewati tugas ini.

Tujuan dari tugas ini adalah untuk membuat basis data RDS. Anda akan menggunakan instans ini di Tugas 3 saat Anda menghubungkannya ke instans EC2 Anda. Langkah-langkah dalam tugas ini mengonfigurasi basis data RDS sebagai berikut:

- Tipe mesin: MySQL
- Templat: Tingkat gratis
- Pengidentifikasi instans DB: **`tutorial-database-manual`**

- Kelas instans DB: `db.t3.micro`

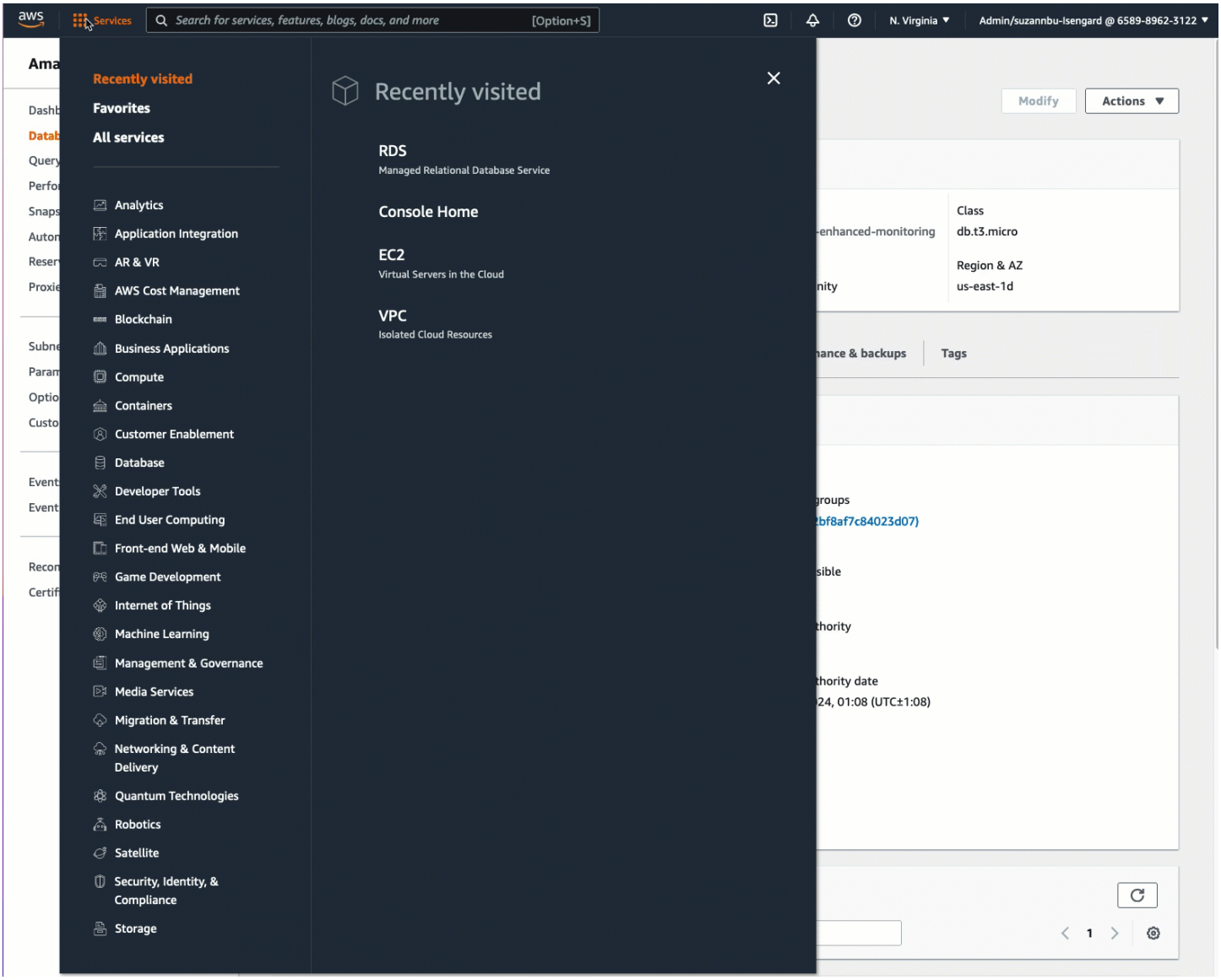
> **Important**
>
> Dalam lingkungan produksi, Anda harus mengonfigurasi instans Anda untuk memenuhi kebutuhan spesifik Anda.

Untuk membuat instans DB MySQL

1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.
2. Dari pemilih Wilayah (di kanan atas), pilih Wilayah AWS tempat Anda membuat instance EC2. Instans EC2 dan instans DB harus berada di Wilayah yang sama.
3. Pada dasbor, pilih Buat basis data.
4. Di bawah Pilih metode pembuatan basid data, pilih Pembuatan mudah. Ketika Anda memilih opsi ini, fitur koneksi otomatis untuk secara otomatis mengonfigurasi koneksi tidak tersedia.
5. Pada Opsi mesin, untuk Tipe mesin, pilih MySQL.
6. Untuk Ukuran instans DB, pilih Tingkat gratis.
7. Untuk Pengidentifikasi instans DB masukkan nama untuk basis data RDS. Untuk tutorial ini, masukkan **`tutorial-database-manual`**.
8. Untuk Nama pengguna master, biarkan nama default, yaitu **`admin`**.
9. Untuk Kata sandi master, masukkan kata sandi yang dapat Anda ingat untuk tutorial ini, kemudian untuk Konfirmasi kata sandi, masukkan kata sandi lagi.
10. Pilih Buat basis data.

    Pada layar Basis Data, Status instans DB baru adalah Membuat sampai instans DB siap digunakan. Saat statusnya berubah menjadi Tersedia, Anda dapat terhubung ke instans DB. Tergantung pada kelas instans DB dan jumlah penyimpanan, diperlukan waktu hingga 20 menit sebelum instans baru tersedia.

Lihat animasi: Membuat instans DB




## Tugas 3: Hubungkan instans EC2 Anda secara manual ke basis data RDS Anda dengan membuat grup keamanan dan menetapkannya ke instans

Tujuan dari tugas ini adalah untuk mereproduksi konfigurasi koneksi dari fitur koneksi otomatis dengan melakukan hal berikut secara manual: Anda membuat dua grup keamanan baru, kemudian menambahkan grup keamanan masing-masing ke instans EC2 dan basis data RDS.

Untuk membuat dua grup keamanan baru dan menetapkan masing-masing ke instans EC2 dan basis data RDS

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Pertama buat grup keamanan untuk ditambahkan ke instans EC2, sebagai berikut:

   a. Pada panel navigasi, pilih Grup Keamanan.

   b. Pilih Buat grup keamanan.

   c. Untuk Nama grup keamanan, masukkan nama deskriptif untuk grup keamanan. Untuk tutorial ini, masukkan **`ec2-rds-manual-configuration`**.

   d. Untuk Deskripsi, masukkan deskripsi singkat. Untuk tutorial ini, masukkan **`EC2 instance security group to allow EC2 instance to securely connect to RDS database`**.

   e. Pilih Buat grup keamanan. Anda akan kembali ke grup keamanan ini untuk menambahkan aturan keluar setelah Anda membuat grup keamanan basis data RDS.

3. Sekarang, buat grup keamanan untuk ditambahkan ke basis data RDS, sebagai berikut:

   a. Pada panel navigasi, pilih Grup Keamanan.

   b. Pilih Buat grup keamanan.

   c. Untuk Nama grup keamanan, masukkan nama deskriptif untuk grup keamanan. Untuk tutorial ini, masukkan **`rds-ec2-manual-configuration`**.

   d. Untuk Deskripsi, masukkan deskripsi singkat. Untuk tutorial ini, masukkan **`RDS database security group to allow EC2 instance to securely connect to RDS database`**.

   e. Pada Aturan masuk, pilih Tambahkan aturan, lalu lakukan hal berikut:

      i. Untuk Jenis, pilih MYSQL/Aurora.

      ii. Untuk Sumber, pilih grup keamanan instans EC2 ec2-rds-manual-configuration yang Anda buat di Langkah 2 prosedur ini.

   f. Pilih Buat grup keamanan.

4. Edit grup keamanan instans EC2 untuk menambahkan aturan keluar, sebagai berikut:

   a. Pada panel navigasi, pilih Grup Keamanan.

   b. Pilih grup keamanan instans EC2 (sebut saja **`ec2-rds-manual-configuration`**), dan pilih tab Aturan keluar.

   c. Pilih Edit aturan keluar.

   d. Pilih Tambahkan aturan, dan lakukan hal-hal berikut:

      i. Untuk Jenis, pilih MYSQL/Aurora.

ii. Untuk Tujuan, pilih grup keamanan database RDS rds-ec2-manual-configuration yang Anda buat pada Langkah 3 prosedur ini.

iii. Pilih Simpan aturan.

5. Tambahkan grup keamanan instans EC2 ke instans EC2 sebagai berikut:

   a. Di panel navigasi, pilih Instans.

   b. Pilih instans EC2 Anda, kemudian pilih Tindakan, Keamanan, Ubah grup keamanan.

   c. Pada Grup keamanan terkait, pilih bidang Pilih grup keamanan, pilih ec2-rds-manual-configuration yang Anda buat sebelumnya, lalu pilih Tambahkan grup keamanan.

   d. Pilih Simpan.

6. Tambahkan grup keamanan basis data RDS ke basis data RDS sebagai berikut:

   a. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.

   b. Di panel navigasi, pilih Basis daya dan plih basis data Anda.

   c. Pilih Ubah.

   d. Di bawah Konektivitas, untuk grup Keamanan, pilih rds-ec2-manual-configuration yang Anda buat sebelumnya, lalu pilih Lanjutkan.

   e. Di bawah Penjadwalan Modifikasi, pilih Terapkan segera.

   f. Pilih Ubah instans DB.

   Anda sekarang telah menyelesaikan langkah manual yang meniru langkah otomatis yang terjadi ketika Anda menggunakan fitur koneksi otomatis.

Anda telah menyelesaikan Opsi 3 dari tutorial ini. Jika Anda telah menyelesaikan Opsi 1, 2, dan 3, dan Anda tidak lagi membutuhkan sumber daya yang dibuat dalam tutorial ini, Anda harus menghapusnya untuk mencegah timbulnya biaya yang tidak perlu. Untuk informasi selengkapnya, lihat Tugas 4 (Opsional): Bersihkan.

## Tugas 4 (Opsional): Bersihkan

> **Warning**
>
> Mengakhiri instance bersifat permanen dan tidak dapat diubah.
> Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk

dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Sekarang setelah Anda menyelesaikan tutorial, itu adalah praktik yang baik untuk membersihkan (menghapus) sumber daya apa pun yang tidak ingin Anda gunakan lagi. Membersihkan AWS sumber daya mencegah akun Anda menimbulkan biaya lebih lanjut.

Jika Anda membuat basis data RDS khusus untuk tutorial ini, Anda dapat mengakhirinya untuk menghentikan biaya apa pun yang terkait dengannya.

Untuk mengakhiri instans menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans yang Anda buat untuk tutorial ini, dan pilih Status instans, Akhiri instans.
4. Pilih Akhiri saat diminta untuk konfirmasi.

Jika Anda membuat basis data RDS khusus untuk tutorial ini, Anda dapat menghapusnya untuk menghentikan biaya apa pun yang terkait dengannya.

Untuk menghapus basis data RDS menggunakan konsol

1. Buka konsol Amazon RDS di https://console.aws.amazon.com/rds/.
2. Di panel navigasi, pilih Basis Data.
3. Pilih basis data RDS yang Anda buat untuk tutorial ini, dan pilih Tindakan, Hapus.
4. Masukkan **`delete me`** di dalam kotak, lalu pilih Hapus.

# Armada EC2 dan Armada Spot

EC2 Fleet dan Spot Fleet dirancang untuk menjadi cara yang berguna untuk meluncurkan armada puluhan, ratusan, atau ribuan instans Amazon EC2 dalam satu operasi. Setiap instance dalam armada dikonfigurasi oleh template peluncuran atau sekumpulan parameter peluncuran yang Anda konfigurasikan secara manual saat peluncuran.

Topik


- Fitur dan manfaat
- Metode armada mana yang terbaik untuk digunakan?
- Opsi konfigurasi untuk Armada EC2 atau Spot Fleet Anda
- Bekerja dengan EC2 Fleet
- Bekerja dengan Spot Fleet
- Pantau Armada EC2 atau Spot Fleet Anda
- Tutorial untuk Armada EC2
- Contoh konfigurasi CLI untuk Armada EC2
- Contoh konfigurasi CLI Spot Fleet
- Kuota untuk Armada EC2 dan Armada Spot


## Fitur dan manfaat

Armada menyediakan fitur dan manfaat berikut, memungkinkan Anda memaksimalkan penghematan biaya dan mengoptimalkan ketersediaan dan kinerja saat menjalankan aplikasi pada beberapa instans EC2.

Beberapa jenis instans

Armada dapat meluncurkan beberapa jenis instans, memastikan tidak bergantung pada ketersediaan jenis instans tunggal. Ini meningkatkan ketersediaan instans secara keseluruhan dalam armada Anda.

Mendistribusikan instans di seluruh Zona Ketersediaan

Armada dapat diluncurkan ke beberapa Zona Ketersediaan, memungkinkan Anda mengurangi biaya dan meningkatkan ketersediaan. Jika armada Anda menyertakan Instans Spot, armada

secara otomatis memilih Zona Ketersediaan berdasarkan preferensi Anda mengenai harga dan gangguan.

Beberapa opsi pembelian

Armada dapat meluncurkan beberapa opsi pembelian (Spot dan On-Demand Instans), memungkinkan Anda mengoptimalkan biaya melalui penggunaan Instans Spot. Anda juga dapat memanfaatkan diskon Instans Cadangan dan Paket Tabungan dengan menggunakannya bersama dengan On-Demand Instans dalam armada.

Penggantian otomatis Instans Spot

Jika armada Anda menyertakan Instans Spot, armada Anda dapat secara otomatis meminta penggantian kapasitas Spot jika Instans Spot Anda terganggu. Melalui Penyeimbangan Kembali Kapasitas, armada juga dapat memantau dan secara proaktif mengganti Instans Spot Anda yang berisiko tinggi mengalami gangguan.

On-Demand Kapasitas cadangan

Armada dapat menggunakan Reservasi On-Demand Kapasitas untuk cadangan On-Demand kapasitas. Armada juga dapat menyertakan Blok Kapasitas untuk ML, memungkinkan Anda untuk memesan instans GPU pada tanggal mendatang untuk mendukung beban kerja pembelajaran mesin (ML) durasi pendek.

## Metode armada mana yang terbaik untuk digunakan?

Sebagai praktik terbaik umum, sebaiknya luncurkan armada Spot dan On-Demand Instans dengan Amazon EC2 Auto Scaling karena menyediakan fitur tambahan yang dapat Anda gunakan untuk mengelola armada Anda. Daftar fitur tambahan termasuk penggantian pemeriksaan kesehatan otomatis untuk Spot dan On-Demand Instans, pemeriksaan kesehatan berbasis aplikasi, dan integrasi dengan Elastic Load Balancing untuk memastikan distribusi lalu lintas aplikasi yang merata ke instans sehat Anda. Anda juga dapat menggunakan grup Penskalaan Otomatis saat menggunakan AWS layanan seperti Amazon ECS, Amazon EKS (grup simpul yang dikelola sendiri), dan Amazon VPC Lattice. Untuk informasi selengkapnya, lihat Panduan Pengguna Amazon EC2 Auto Scaling.

Jika Anda tidak dapat menggunakan Amazon EC2 Auto Scaling, maka Anda dapat mempertimbangkan untuk menggunakan EC2 Fleet atau Spot Fleet. EC2 Fleet dan Spot Fleet menawarkan fungsionalitas inti yang sama. Namun, EC2 Fleet hanya tersedia menggunakan baris

perintah dan tidak memberikan dukungan konsol. Spot Fleet menyediakan dukungan konsol, tetapi menggunakan API lama tanpa investasi yang direncanakan.

Gunakan tabel berikut untuk menentukan metode armada mana yang akan digunakan.

| Metode armada | Kapan harus menggunakan? | Kasus penggunaan |
| --- | --- | --- |
| Amazon EC2 Auto Scaling | • Anda memerlukan beberapa instans dengan konfigurasi tunggal atau konfigurasi campuran.<br>• Anda ingin mengotoma tiskan manajemen siklus hidup instans Anda. | Buat grup Auto Scaling yang mengelola siklus hidup instans Anda sambil mempertahankan jumlah instans yang diinginka n. Mendukung penskalaan horizontal (menambahkan lebih banyak instans) antara batas minimum dan maksimum yang ditentukan. |
| Armada EC2 | • Anda memerlukan beberapa instans dengan konfigurasi tunggal atau konfigurasi campuran.<br>• Anda ingin mengelola sendiri siklus hidup instans Anda.<br>• Jika Anda tidak memerluka n penskalaan otomatis, sebaiknya gunakan `instant` tipe EC2 Fleet. | Buat `instant` armada Instans dan On-Demand Instans Spot dalam satu operasi, dengan beberapa spesifikasi peluncura n yang bervariasi menurut jenis instans, AMI, Zona Ketersediaan, atau subnet. Strategi alokasi Instans Spot secara default `lowest-pr ice` adalah per unit, tetapi kami sarankan mengubahnya menjadi`price-capacity-optimized` . |
| Armada Spot | • Kami sangat tidak menyarankan menggunak an Spot Fleet karena menggunakan API lama tanpa investasi yang direncanakan. | Gunakan Spot Fleet hanya jika Anda memerlukan dukungan konsol untuk kasus penggunaa n saat Anda akan menggunak an EC2 Fleet. |

| Metode armada | Kapan harus menggunakan? | Kasus penggunaan |
| --- | --- | --- |
| | • Jika Anda ingin mengelola siklus hidup instans, gunakan EC2 Fleet.<br>• Jika Anda tidak ingin mengelola siklus hidup instans, gunakan grup Penskalaan Otomatis. | |

## Opsi konfigurasi untuk Armada EC2 atau Spot Fleet Anda

Saat merencanakan Armada EC2 atau Spot Fleet Anda, kami sarankan Anda mempertimbangkan opsi berikut saat memutuskan cara mengonfigurasi armada Anda.

| Opsi Konfigurasi | Pertanyaan | Dokumentasi |
| --- | --- | --- |
| Jenis permintaan armada | Apakah Anda menginginkan armada yang mengajukan permintaan satu kali untuk kapasitas target yang diinginkan, atau armada yang mempertahankan kapasitas target dari waktu ke waktu? | Jenis permintaan Armada dan Armada Spot EC2 |
| Instans Spot | Apakah Anda berencana untuk menyertakan Instans Spot dalam armada Anda? Tinjau praktik terbaik Spot dan gunakan saat merencanakan armada Anda sehingga Anda dapat menyediakan instans dengan harga serendah mungkin. | Praktik terbaik untuk Amazon EC2 Spot |
| Batas pengeluaran untuk armada Anda | Apakah Anda ingin membatasi berapa banyak yang akan Anda bayar untuk armada Anda per jam? | Tetapkan batas pengeluaran untuk Armada EC2 atau Spot Fleet Anda |

| Opsi Konfigurasi | Pertanyaan | Dokumentasi |
| --- | --- | --- |
| Jenis instans dan pemilihan tipe instance berbasis atribut | Apakah Anda ingin menentukan jenis instans di armada Anda, atau membiarkan Amazon EC2 memilih jenis instans yang memenuhi persyaratan aplikasi Anda? | Tentukan atribut untuk pemilihan tipe instans untuk EC2 Fleet atau Spot Fleet |
| Pembobotan instans | Apakah Anda ingin menetapkan bobot untuk setiap jenis instans untuk mewakili kapasitas dan kinerja komputasi mereka, sehingga Amazon EC2 dapat memilih kombinasi jenis instans yang tersedia untuk memenuhi kapasitas target yang Anda inginkan? | Gunakan pembobotan instans untuk mengelola biaya dan kinerja Armada EC2 atau Spot Fleet |
| Strategi alokasi | Apakah Anda ingin memutuskan apakah akan mengoptimalkan kapasitas, harga, atau jenis instans yang tersedia untuk Instans Spot dan On-Demand Instans di armada Anda? | Gunakan strategi alokasi untuk menentukan bagaimana EC2 Fleet atau Spot Fleet memenuhi Spot dan kapasitas On-Demand |
| Penyeimbangan Ulang Kapasitas | Apakah Anda ingin armada Anda secara otomatis menggantikan Instans Spot yang berisiko? | Gunakan Penyeimbangan Kembali Kapasitas di Armada EC2 dan Spot Fleet untuk menggantikan Instans Spot yang berisiko |
| On-Demand Reservasi Kapasitas | Apakah Anda ingin mencadangkan kapasitas untuk On-Demand Instans di armada Anda? | Gunakan Reservasi Kapasitas untuk cadangan On-Demand kapasitas di Armada EC2 |

| Opsi Konfigurasi | Pertanyaan | Dokumentasi |
| --- | --- | --- |
| Beberapa Jenis Reservasi Kapasitas | Apakah Anda ingin meluncurkan instans di beberapa jenis Reservasi Kapasitas — Reservasi On-Demand Kapasitas, Blok Kapasitas untuk ML, dan Reservasi Kapasitas yang dapat diinterupsi — menggunakan Grup Sumber Daya Reservasi Kapasitas tunggal, dan secara opsional kembali ke On-Demand kapasitas? | Tutorial: Mengkonfigurasi Armada EC2 Anda untuk meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas |

## Jenis permintaan Armada dan Armada Spot EC2

> **Note**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Jenis permintaan untuk Armada EC2 atau Armada Spot menentukan apakah permintaan tersebut sinkron atau asinkron, dan apakah itu permintaan satu kali untuk kapasitas target yang diinginkan atau upaya berkelanjutan untuk mempertahankan kapasitas dari waktu ke waktu. Saat mengonfigurasi armada Anda, Anda harus menentukan jenis permintaan.

Baik EC2 Fleet dan Spot Fleet menawarkan dua jenis permintaan: `request` dan`maintain`. Selain itu, EC2 Fleet menawarkan jenis permintaan ketiga yang disebut`instant`.

Jenis permintaan armada

`instant`(Hanya Armada EC2)

Jika Anda mengonfigurasi tipe permintaan sebagai `instant`, Armada EC2 akan mengajukan permintaan satu kali sinkron untuk kapasitas yang Anda inginkan. Dalam respons API, ia mengembalikan instance yang diluncurkan dan memberikan kesalahan untuk instans yang tidak dapat diluncurkan. Untuk informasi selengkapnya, lihat Konfigurasikan jenis Armada EC2 instant.

`request`

Jika Anda mengonfigurasi jenis permintaan sebagai`request`, armada menempatkan permintaan satu kali asinkron untuk kapasitas yang Anda inginkan. Jika kapasitas berkurang karena gangguan Spot, armada tidak mencoba untuk mengisi kembali Instans Spot, juga tidak mengirimkan permintaan di kumpulan kapasitas Spot alternatif jika kapasitas tidak tersedia. Saat membuat jenis Spot Fleet `request` menggunakan konsol, kosongkan kotak centang Per tahankan kapasitas target.

`maintain` (default)

Jika Anda mengonfigurasi jenis permintaan sebagai`maintain`, armada menempatkan permintaan asinkron untuk kapasitas yang Anda inginkan, dan mempertahankannya dengan mengisi kembali Instans Spot yang terputus secara otomatis. Saat membuat jenis Spot Fleet `maintain` menggunakan konsol, pilih kotak centang Pertahankan kapasitas target

## Konfigurasikan jenis Armada EC2 instant

Armada EC2 tipe instan adalah permintaan satu kali sinkron yang hanya membuat satu upaya untuk meluncurkan kapasitas yang Anda inginkan. Respons API mencantumkan instans yang diluncurkan, bersama dengan kesalahan untuk instans yang tidak dapat diluncurkan. Terdapat beberapa keuntungan dari penggunaan Armada EC2 tipe instan, yang dijelaskan dalam artikel ini. Contoh konfigurasi disediakan di akhir artikel.

Untuk beban kerja yang memerlukan API khusus peluncuran untuk meluncurkan instans EC2, Anda dapat menggunakan API. RunInstances Namun, dengan RunInstances, Anda hanya dapat meluncurkan Instans Sesuai Permintaan atau Instans Spot, tetapi tidak keduanya dalam permintaan yang sama. Selanjutnya, ketika Anda menggunakan RunInstances untuk meluncurkan Instans Spot, permintaan Instans Spot Anda terbatas pada satu jenis instans dan satu Availability Zone. API ini menargetkan kolam kapasitas Spot (set instans yang tidak digunakan dengan tipe instans dan Zona ketersediaan yang sama). Jika kumpulan kapasitas Spot tidak memiliki kapasitas Instans Spot yang memadai untuk permintaan Anda, RunInstances panggilan gagal.

Alih-alih menggunakan RunInstances untuk meluncurkan Instans Spot, sebaiknya gunakan CreateFleet API dengan `type` parameter yang disetel `instant` untuk manfaat berikut:

- Luncurkan Instans Sesuai Permintaan dan Instans Spot dalam satu permintaan. Armada EC2 dapat meluncurkan Instans Sesuai Permintaan, Instans Spot, atau keduanya. Permintaan

Instans Spot terpenuhi jika terdapat kapasitas yang tersedia dan harga maksimum per jam untuk permintaan Anda melebihi harga Spot.

- Menambah ketersediaan Instans Spot. Dengan menggunakan Armada EC2 tipe `instant`, Anda dapat meluncurkan Instans Spot mengikuti praktik terbaik Spot dengan keuntungan yang dihasilkan:
  - Praktik terbaik Spot: Bersikaplah fleksibel terkait tipe instans dan Zona Ketersediaan.

    Keuntungan: Dengan menentukan beberapa tipe instans dan Zona Ketersediaan, Anda menambah jumlah kolam kapasitas Spot. Hal ini memberi layanan Spot kesempatan yang lebih baik untuk menemukan dan mengalokasikan kapasitas komputasi Spot yang Anda inginkan. Praktik terbaiknya adalah fleksibel pada setidaknya 10 tipe instans untuk setiap beban kerja dan memastikan semua Zona Ketersediaan dikonfigurasi untuk digunakan di VPC Anda.
  - Spot best practice: Gunakan strategi price-capacity-optimized alokasi.

    Manfaat: Strategi `price-capacity-optimized` alokasi mengidentifikasi instance dari kumpulan kapasitas Spot yang paling tersedia, dan kemudian secara otomatis menyediakan instance dari harga terendah dari kumpulan ini. Karena kapasitas Instans Spot Anda bersumber dari kolam dengan kapasitas optimal, hal ini mengurangi kemungkinan bahwa Instans Spot Anda akan terinterupsi saat Amazon EC2 membutuhkan kapasitas kembali.
- Dapatkan akses ke set kemampuan yang lebih luas. Untuk beban kerja yang memerlukan API khusus peluncuran, dan di mana Anda lebih suka mengelola siklus hidup instans Anda daripada membiarkan Armada EC2 mengelolanya untuk Anda, gunakan jenis Armada EC2 alih-alih API. `instant` RunInstances Armada EC2 menyediakan serangkaian kemampuan yang lebih luas daripada RunInstances, seperti yang ditunjukkan dalam contoh berikut. Untuk semua beban kerja lainnya, Anda harus menggunakan Amazon EC2 Auto Scaling karena menyediakan set fitur yang lebih komprehensif untuk berbagai macam beban kerja, seperti aplikasi yang didukung ELB, beban kerja terkontainerisasi, dan tugas pemrosesan antrean.

Anda dapat menggunakan Armada EC2 tipe instan untuk meluncurkan instans ke Blok Kapasitas. Untuk informasi selengkapnya, lihat Tutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Blok Kapasitas.

AWS layanan seperti Amazon EC2 Auto Scaling dan Amazon EMR menggunakan EC2 Fleet tipe instan untuk meluncurkan instans EC2.

### Prasyarat untuk Armada EC2 tipe instan

Untuk prasyarat guna membuat Armada EC2, lihat Prasyarat Armada EC2.

### Cara kerja Armada EC2

Saat bekerja dengan Armada EC2 tipe `instant`, urutan peristiwanya adalah sebagai berikut:

1. Konfigurasi: Konfigurasikan jenis CreateFleetpermintaan sebagai`instant`. Untuk informasi selengkapnya, lihat Membuat Armada EC2. Perhatikan bahwa setelah melakukan panggilan API, Anda tidak dapat memodifikasinya.
2. Permintaan: Saat Anda melakukan panggilan API, Amazon EC2 menempatkan permintaan satu kali sinkron untuk kapasitas yang Anda inginkan.
3. Respons: Respons API mencantumkan instance yang diluncurkan, bersama dengan kesalahan untuk instance yang tidak dapat diluncurkan.
4. Jelaskan: Anda dapat mendeskripsikan Armada EC2 Anda, daftar instans yang terkait dengan Armada EC2 Anda, dan melihat riwayat Armada EC2 Anda.
5. Mengakhiri instance: Anda dapat menghentikan instance kapan saja.
6. Hapus permintaan armada: Permintaan armada dapat dihapus secara manual atau otomatis:
   - Petunjuk: Anda dapat menghapus permintaan armada setelah instans diluncurkan.

     Perhatikan bahwa `instant` armada yang dihapus dengan instance yang sedang berjalan tidak didukung. Saat Anda menghapus `instant` armada, Amazon EC2 secara otomatis menghentikan semua instance-nya. Untuk armada dengan lebih dari 1000 instance, permintaan penghapusan mungkin gagal. Jika armada Anda memiliki lebih dari 1000 instans, pertama-tama hentikan sebagian besar instans secara manual, sisakan 1000 atau kurang. Kemudian hapus armada, dan instance yang tersisa akan dihentikan secara otomatis.
   - Otomatis: Amazon EC2 menghapus permintaan armada beberapa saat setelah:
     - Semua instance dihentikan.
     - Armada gagal meluncurkan instance apa pun.

### Contoh

Contoh berikut menunjukkan cara menggunakan Armada EC2 tipe `instant` pada berbagai kasus penggunaan. Untuk informasi selengkapnya tentang penggunaan parameter CreateFleet API EC2, lihat CreateFleetdi Referensi API Amazon EC2.

Contoh


- Contoh 1: Meluncurkan Instans Spot dengan strategi alokasi yang dioptimalkan kapasitas
- Contoh 2: Meluncurkan satu Instans Spot dengan strategi alokasi yang dioptimalkan kapasitas
- Contoh 3: Meluncurkan Instans Spot menggunakan pembobotan instans
- Contoh 4: Luncurkan Instans Spot dalam Zona Ketersediaan tunggal
- Contoh 5: Meluncurkan Instans Spot satu tipe instans dalam satu Zona Ketersediaan
- Contoh 6: Meluncurkan Instans Spot hanya jika kapasitas target minimum dapat diluncurkan
- Contoh 7: Meluncurkan Instans Spot hanya jika kapasitas target minimum dapat diluncurkan dari Tipe Instans yang sama dalam satu Zona Ketersediaan
- Contoh 8: Meluncurkan instans dengan banyak Templat Peluncuran
- Contoh 9: Meluncurkan Instans Spot dengan basis Instans Sesuai Permintaan
- Contoh 10: Meluncurkan Instans Spot menggunakan strategi alokasi yang dioptimalkan kapasitas dengan basis Instans Sesuai Permintaan menggunakan Reservasi Kapasitas dan strategi alokasi yang diprioritaskan
- Contoh 11: Luncurkan Instans Spot menggunakan strategi capacity-optimized-prioritized alokasi
- Contoh 12: Menentukan parameter Systems Manager, bukan ID AMI


Contoh 1: Meluncurkan Instans Spot dengan strategi alokasi yang dioptimalkan kapasitas

Contoh berikut menentukan parameter yang diperlukan dalam Armada EC2 tipe `instant`: templat peluncuran, kapasitas target, opsi pembelian default, dan penimpaan templat peluncuran.

- Templat peluncuran diidentifikasi dengan nama templat dan nomor versi peluncurannya.
- 12 penggantian templat peluncuran menentukan 4 tipe instans yang berbeda dan 3 subnet berbeda, masing-masing di Zona Ketersediaan terpisah. Setiap tipe instans dan kombinasi subnet menentukan kolam kapasitas Spot, sehingga menghasilkan 12 kolam kapasitas Spot.
- Kapasitas target untuk armada adalah 20 instans.
- Opsi pembelian default adalah `spot`, yang menghasilkan armada yang berupaya meluncurkan 20 Instans Spot ke kolam kapasitas Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
```

```
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-49e41922"
            },
```

```
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 2: Meluncurkan satu Instans Spot dengan strategi alokasi yang dioptimalkan kapasitas

Anda dapat meluncurkan satu Instans Spot secara optimal pada satu waktu dengan melakukan beberapa jenis panggilan API Armada EC2`instant`, dengan menyetel TotalTargetCapacity ke 1.

Contoh berikut menentukan parameter yang diperlukan dalam Armada EC2 tipe instan: templat peluncuran, kapasitas target, opsi pembelian default, dan penimpaan templat peluncuran. Templat peluncuran diidentifikasi dengan nama templat dan nomor versi peluncurannya. 12 penimpaan templat peluncuran memiliki 4 tipe instans yang berbeda dan 3 subnet yang berbeda, masing-masing di Zona Ketersediaan yang terpisah. Kapasitas target untuk armada adalah 1 instans, dan opsi pembelian default adalah spot, yang mengakibatkan armada berupaya meluncurkan Instans Spot dari salah satu dari 12 kolam kapasitas Spot berdasarkan strategi alokasi yang dioptimalkan kapasitas, untuk meluncurkan Instans Spot dari kolam kapasitas dengan ketersediaan paling tinggi.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "LaunchTemplateConfigs": [
      {
```

```
        "LaunchTemplateSpecification":{
           "LaunchTemplateName":"ec2-fleet-lt1",
           "Version":"$Latest"
        },
        "Overrides":[
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-fae8c380"
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-e7188bab"
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-49e41922"
           },
           {
              "InstanceType":"m5d.large",
              "SubnetId":"subnet-fae8c380"
```

```
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 1,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 3: Meluncurkan Instans Spot menggunakan pembobotan instans

Contoh berikut menggunakan pembobotan instans, yang berarti harga adalah per unit jam, bukan per jam instans. Setiap konfigurasi peluncuran mencantumkan jenis instans yang berbeda dan bobot yang berbeda berdasarkan berapa banyak unit beban kerja yang dapat dijalankan pada instance dengan asumsi unit beban kerja memerlukan memori 15 GB dan 4 v. CPUs Misalnya m5.xlarge (memori 4 v CPUs dan 16 GB) dapat menjalankan satu unit dan berbobot 1, m5.2xlarge (8 v CPUs dan 32 GB memori) dapat menjalankan 2 unit dan berbobot 2, dan seterusnya. Total kapasitas target diatur ke 40 unit. Opsi pembelian default adalah spot, dan strategi alokasi dioptimalkan kapasitas, yang menghasilkan 40 m5.xlarge (40 dibagi 1), 20 m5.2xlarge (40 dibagi 2), 10 m5.4xlarge (40 dibagi 4), 5 m5.8xlarge (40 dibagi 8), atau campuran tipe instans dengan bobot yang ditambahkan ke kapasitas yang diinginkan berdasarkan kapasitas strategi alokasi yang dioptimalkan kapasitas.

Untuk informasi selengkapnya, lihat Gunakan pembobotan instans untuk mengelola biaya dan kinerja Armada EC2 atau Spot Fleet.

```
{
   "SpotOptions":{
      "AllocationStrategy":"capacity-optimized"
   },
   "LaunchTemplateConfigs":[
      {
         "LaunchTemplateSpecification":{
```

```
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"m5.xlarge",
               "SubnetId":"subnet-fae8c380",
               "WeightedCapacity":1
            },
            {
               "InstanceType":"m5.xlarge",
               "SubnetId":"subnet-e7188bab",
               "WeightedCapacity":1
            },
            {
               "InstanceType":"m5.xlarge",
               "SubnetId":"subnet-49e41922",
               "WeightedCapacity":1
            },
            {
               "InstanceType":"m5.2xlarge",
               "SubnetId":"subnet-fae8c380",
               "WeightedCapacity":2
            },
            {
               "InstanceType":"m5.2xlarge",
               "SubnetId":"subnet-e7188bab",
               "WeightedCapacity":2
            },
            {
               "InstanceType":"m5.2xlarge",
               "SubnetId":"subnet-49e41922",
               "WeightedCapacity":2
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-fae8c380",
               "WeightedCapacity":4
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-e7188bab",
               "WeightedCapacity":4
            },
```

```
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-49e41922",
               "WeightedCapacity":4
            },
            {
               "InstanceType":"m5.8xlarge",
               "SubnetId":"subnet-fae8c380",
               "WeightedCapacity":8
            },
            {
               "InstanceType":"m5.8xlarge",
               "SubnetId":"subnet-e7188bab",
               "WeightedCapacity":8
            },
            {
               "InstanceType":"m5.8xlarge",
               "SubnetId":"subnet-49e41922",
               "WeightedCapacity":8
            }
         ]
      }
   ],
   "TargetCapacitySpecification":{
      "TotalTargetCapacity":40,
      "DefaultTargetCapacityType":"spot"
   },
   "Type":"instant"
}
```

Contoh 4: Luncurkan Instans Spot dalam Zona Ketersediaan tunggal

Anda dapat mengonfigurasi armada untuk meluncurkan semua instance dalam satu Availability Zone dengan menyetel opsi Spot SingleAvailabilityZone ke true.

12 penimpaan templat peluncuran memiliki tipe dan subnet instans yang berbeda (masing–masing di Zona Ketersediaan terpisah), tetapi kapasitas tertimbangnya sama. Total kapasitas target adalah 20 instans, opsi pembelian default adalah spot, dan strategi alokasi Spot dioptimalkan kapasitas. Armada EC2 meluncurkan 20 Instans Spot semuanya dalam satu AZ, dari kolam kapasitas Spot dengan kapasitas optimal menggunakan spesifikasi peluncuran.

```
{
    "SpotOptions": {
```

```
        "AllocationStrategy": "capacity-optimized",
        "SingleAvailabilityZone": true
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5.4xlarge",
```

```
                "SubnetId":"subnet-49e41922"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 5: Meluncurkan Instans Spot satu tipe instans dalam satu Zona Ketersediaan

Anda dapat mengonfigurasi armada untuk meluncurkan semua instance dari jenis instans yang sama dan dalam Availability Zone tunggal dengan menyetel SpotOptions SingleInstanceType ke true dan SingleAvailabilityZone true.

12 penimpaan templat peluncuran memiliki tipe dan subnet instans yang berbeda (masing–masing di Zona Ketersediaan terpisah), tetapi kapasitas tertimbangnya sama. Total kapasitas target adalah 20 instans, opsi pembelian default adalah spot, strategi alokasi Spot dioptimalkan kapasitas. Armada EC2 meluncurkan 20 Instans Spot dengan tipe instans yang sama, semuanya dalam satu AZ dari kolam Instans Spot dengan kapasitas optimal menggunakan spesifikasi peluncuran.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true
    },
    "LaunchTemplateConfigs": [
```

```
    {
       "LaunchTemplateSpecification":{
          "LaunchTemplateName":"ec2-fleet-lt1",
          "Version":"$Latest"
       },
       "Overrides":[
          {
             "InstanceType":"c5.4xlarge",
             "SubnetId":"subnet-fae8c380"
          },
          {
             "InstanceType":"c5.4xlarge",
             "SubnetId":"subnet-e7188bab"
          },
          {
             "InstanceType":"c5.4xlarge",
             "SubnetId":"subnet-49e41922"
          },
          {
             "InstanceType":"c5d.4xlarge",
             "SubnetId":"subnet-fae8c380"
          },
          {
             "InstanceType":"c5d.4xlarge",
             "SubnetId":"subnet-e7188bab"
          },
          {
             "InstanceType":"c5d.4xlarge",
             "SubnetId":"subnet-49e41922"
          },
          {
             "InstanceType":"m5.4xlarge",
             "SubnetId":"subnet-fae8c380"
          },
          {
             "InstanceType":"m5.4xlarge",
             "SubnetId":"subnet-e7188bab"
          },
          {
             "InstanceType":"m5.4xlarge",
             "SubnetId":"subnet-49e41922"
          },
          {
             "InstanceType":"m5d.4xlarge",
```

```
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 6: Meluncurkan Instans Spot hanya jika kapasitas target minimum dapat diluncurkan

Anda dapat mengonfigurasi armada untuk meluncurkan instance hanya jika kapasitas target minimum dapat diluncurkan dengan menyetel opsi Spot MinTargetCapacity ke kapasitas target minimum yang ingin Anda luncurkan bersama.

Saat menentukan MinTargetCapacity, Anda harus menentukan setidaknya satu dari parameter ini: SingleInstanceType atau SingleAvailabilityZone. Dalam contoh ini, SingleInstanceType ditentukan, sehingga semua 20 instance harus menggunakan jenis instance yang sama.

12 penimpaan templat peluncuran memiliki tipe dan subnet instans yang berbeda (masing–masing di Zona Ketersediaan terpisah), tetapi kapasitas tertimbangnya sama. Total kapasitas target dan kapasitas target minimum keduanya disetel ke 20 instance, opsi pembelian default adalah spot, dan strategi alokasi Spot dioptimalkan kapasitas. Armada EC2 meluncurkan 20 Instans Spot dari kolam kapasitas Spot dengan kapasitas optimal menggunakan penimpaan templat peluncuran, hanya jika dapat meluncurkan semua 20 instans pada saat yang bersamaan.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "SingleInstanceType": true,
        "MinTargetCapacity": 20
```

```
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
```

```
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"m5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 7: Meluncurkan Instans Spot hanya jika kapasitas target minimum dapat diluncurkan dari Tipe Instans yang sama dalam satu Zona Ketersediaan

Anda dapat mengonfigurasi armada untuk meluncurkan instance hanya jika kapasitas target minimum dapat diluncurkan dengan satu jenis instans dalam satu Availability Zone dengan menyetel opsi Spot MinTargetCapacity ke kapasitas target minimum yang ingin Anda luncurkan bersama SingleInstanceType dan SingleAvailabilityZone opsi.

12 spesifikasi peluncuran yang menimpa templat peluncuran, memiliki tipe instans dan subnet yang berbeda (masing-masing di Zona Ketersediaan yang terpisah), tetapi kapasitas tertimbangnya sama. Total kapasitas target dan kapasitas target minimum keduanya disetel ke 20 instance, opsi pembelian default adalah spot, strategi alokasi Spot dioptimalkan kapasitas, benar dan benar SingleInstanceType . SingleAvailabilityZone Armada EC2 meluncurkan 20 Instans Spot dengan tipe instans yang sama, semuanya dalam satu AZ dari kolam kapasitas Spot dengan kapasitas optimal menggunakan spesifikasi peluncuran, hanya jika dapat meluncurkan semua 20 instans pada saat yang bersamaan.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
```

```
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true,
        "MinTargetCapacity": 20
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.4xlarge",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"m5.4xlarge",
               "SubnetId":"subnet-e7188bab"
            },
            {
```

```
                "InstanceType":"m5.4xlarge",
                "SubnetId":"subnet-49e41922"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"m5d.4xlarge",
                "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 8: Meluncurkan instans dengan banyak Templat Peluncuran

Anda dapat mengonfigurasi armada untuk meluncurkan instans dengan spesifikasi peluncuran yang berbeda untuk berbagai tipe instans atau grup tipe instans, dengan menentukan banyak templat peluncuran. Dalam contoh ini kita ingin agar memiliki ukuran volume EBS yang berbeda untuk tipe instans yang berbeda dan dikonfigurasi dalam templat peluncuran ec2-fleet-lt-4xl, ec2-fleet-lt-9xl, dan ec2-fleet-lt-18xl.

Dalam contoh ini, kita menggunakan 3 templat peluncuran yang berbeda untuk 3 tipe instans berdasarkan ukurannya. Spesifikasi peluncuran yang ditimpa pada semua templat peluncuran menggunakan bobot instans berdasarkan vCPU pada tipe instans. Total kapasitas target adalah 144 instans, opsi pembelian default adalah spot, dan strategi alokasi Spot dioptimalkan kapasitas. Armada EC2 dapat meluncurkan 9 c5n.4xlarge (144 dibagi 16) menggunakan templat peluncuran ec2-fleet-4xl atau 4 c5n.9xlarge (144 dibagi 36) menggunakan template peluncuran ec2-fleet-9xl, atau 2 c5n.18xlarge (144 dibagi 72) menggunakan templat peluncuran ec2-fleet-18xl, atau campuran

tipe instans dengan bobot yang ditambahkan ke kapasitas yang diinginkan berdasarkan strategi alokasi yang dioptimalkan kapasitas.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt-18xl",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5n.18xlarge",
               "SubnetId":"subnet-fae8c380",
               "WeightedCapacity":72
            },
            {
               "InstanceType":"c5n.18xlarge",
               "SubnetId":"subnet-e7188bab",
               "WeightedCapacity":72
            },
            {
               "InstanceType":"c5n.18xlarge",
               "SubnetId":"subnet-49e41922",
               "WeightedCapacity":72
            }
         ]
      },
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt-9xl",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5n.9xlarge",
               "SubnetId":"subnet-fae8c380",
               "WeightedCapacity":36
            },
            {
```

```
                "InstanceType":"c5n.9xlarge",
                "SubnetId":"subnet-e7188bab",
                "WeightedCapacity":36
             },
             {
                "InstanceType":"c5n.9xlarge",
                "SubnetId":"subnet-49e41922",
                "WeightedCapacity":36
             }
          ]
       },
       {
          "LaunchTemplateSpecification":{
             "LaunchTemplateName":"ec2-fleet-lt-4xl",
             "Version":"$Latest"
          },
          "Overrides":[
             {
                "InstanceType":"c5n.4xlarge",
                "SubnetId":"subnet-fae8c380",
                "WeightedCapacity":16
             },
             {
                "InstanceType":"c5n.4xlarge",
                "SubnetId":"subnet-e7188bab",
                "WeightedCapacity":16
             },
             {
                "InstanceType":"c5n.4xlarge",
                "SubnetId":"subnet-49e41922",
                "WeightedCapacity":16
             }
          ]
       }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 144,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 9: Meluncurkan Instans Spot dengan basis Instans Sesuai Permintaan

Contoh berikut menentukan total kapasitas target dari 20 instans untuk armada tersebut dan kapasitas target dari 5 Instans Sesuai Permintaan. Opsi pembelian default adalah spot. Armada meluncurkan 5 Instans Sesuai Permintaan sebagaimana ditentukan, tetapi perlu meluncurkan 15 instans lagi untuk memenuhi total kapasitas target. Opsi pembelian untuk selisih dihitung sebagai TotalTargetCapacity — OnDemandTargetCapacity = DefaultTargetCapacityType, yang menghasilkan peluncuran armada 15 Instans Spot membentuk salah satu dari 12 kumpulan kapasitas Spot berdasarkan strategi alokasi yang dioptimalkan kapasitas.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "LaunchTemplateConfigs": [
      {
         "LaunchTemplateSpecification":{
            "LaunchTemplateName":"ec2-fleet-lt1",
            "Version":"$Latest"
         },
         "Overrides":[
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5.large",
               "SubnetId":"subnet-49e41922"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-fae8c380"
            },
            {
               "InstanceType":"c5d.large",
               "SubnetId":"subnet-e7188bab"
            },
            {
               "InstanceType":"c5d.large",
```

```
                "SubnetId":"subnet-49e41922"
            },
            {
                "InstanceType":"m5.large",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"m5.large",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"m5.large",
                "SubnetId":"subnet-49e41922"
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-fae8c380"
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-e7188bab"
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-49e41922"
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "OnDemandTargetCapacity": 5,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 10: Meluncurkan Instans Spot menggunakan strategi alokasi yang dioptimalkan kapasitas dengan basis Instans Sesuai Permintaan menggunakan Reservasi Kapasitas dan strategi alokasi yang diprioritaskan

Anda dapat mengonfigurasi armada untuk menggunakan Reservasi Kapasitas Sesuai Permintaan terlebih dahulu saat meluncurkan basis Instans Sesuai Permintaan dengan tipe kapasitas target

default sebagai spot dengan menyetel strategi penggunaan untuk Reservasi Kapasitas. use-capacity-reservations-first Jika lebih dari satu kolam instans memiliki Reservasi Kapasitas yang tidak terpakai, strategi alokasi Sesuai Permintaan akan diterapkan. Dalam contoh ini, strategi alokasi Sesuai Permintaan diprioritaskan.

Dalam contoh ini, terdapat 6 Reservasi Kapasitas yang tidak terpakai yang tersedia. Jumlah tersebut kurang dari kapasitas Sesuai Permintaan target armada 10 instans Sesuai Permintaan.

Akun tersebut memiliki 6 Reservasi Kapasitas yang tidak terpakai dalam 2 kolam. Jumlah Cadangan Kapasitas di setiap kumpulan ditunjukkan dengan AvailableInstanceCount.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

Konfigurasi armada berikut hanya menampilkan konfigurasi terkait untuk contoh ini. Strategi alokasi On-Demand diprioritaskan, dan strategi penggunaan untuk Reservasi Kapasitas adalah. use-capacity-reservations-first Strategi alokasi Spot dioptimalkan kapasitas. Total kapasitas target adalah 20, kapasitas target Sesuai Permintaan adalah 10, dan tipe kapasitas target default adalah spot.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized"
    },
    "OnDemandOptions":{
       "CapacityReservationOptions": {
```

```
        "UsageStrategy": "use-capacity-reservations-first"
      },
      "AllocationStrategy":"prioritized"
   },
   "LaunchTemplateConfigs": [
     {
        "LaunchTemplateSpecification":{
           "LaunchTemplateName":"ec2-fleet-lt1",
           "Version":"$Latest"
        },
        "Overrides":[
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-fae8c380",
              "Priority": 1.0
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-e7188bab",
              "Priority": 2.0
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-49e41922",
              "Priority": 3.0
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-fae8c380",
              "Priority": 4.0
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-e7188bab",
              "Priority": 5.0
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-49e41922",
              "Priority": 6.0
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-fae8c380",
```

```
                "Priority": 7.0
            },
            {
                "InstanceType":"m5.large",
                "SubnetId":"subnet-e7188bab",
                "Priority": 8.0
            },
            {
                "InstanceType":"m5.large",
                "SubnetId":"subnet-49e41922",
                "Priority": 9.0
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-fae8c380",
                "Priority": 10.0
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-e7188bab",
                "Priority": 11.0
            },
            {
                "InstanceType":"m5d.large",
                "SubnetId":"subnet-49e41922",
                "Priority": 12.0
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "OnDemandTargetCapacity": 10,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Setelah Anda membuat armada instan menggunakan konfigurasi sebelumnya, 20 instans berikut ini diluncurkan untuk memenuhi kapasitas target:

- 7 Instans Sesuai Permintaan c5.large di us-east-1a – c5.large di us-east-1a diprioritaskan terlebih dahulu, dan terdapat 3 Reservasi Kapasitas c5.large yang tidak terpakai. Reservasi Kapasitas

digunakan terlebih dahulu untuk meluncurkan 3 Instans Sesuai Permintaan dan 4 Instans Sesuai Permintaan tambahan diluncurkan sesuai dengan strategi alokasi Sesuai Permintaan, yang diprioritaskan dalam contoh ini.

- 3 m5.large Instans Sesuai Permintaan di us-east-1a — m5.large di us-east-1a diprioritaskan kedua, dan ada 3 Reservasi Kapasitas m5.large yang tidak terpakai yang tersedia.
- 10 Instans Spot dari salah satu dari 12 kolam kapasitas Spot yang memiliki kapasitas optimal sesuai dengan strategi alokasi yang dioptimalkan kapasitas.

Setelah armada diluncurkan, Anda dapat berlari describe-capacity-reservationsuntuk melihat berapa banyak Reservasi Kapasitas yang tidak terpakai yang tersisa. Dalam contoh ini, Anda akan melihat respons berikut, yang menunjukkan bahwa semua Reservasi Kapasitas c5.large dan m5.large telah digunakan.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.large",
    "AvailableInstanceCount": 0
}
```

Contoh 11: Luncurkan Instans Spot menggunakan strategi capacity-optimized-prioritized alokasi

Contoh berikut menentukan parameter yang diperlukan dalam Armada EC2 tipe instan: templat peluncuran, kapasitas target, opsi pembelian default, dan penimpaan templat peluncuran. Templat peluncuran diidentifikasi dengan nama templat dan nomor versi peluncurannya. 12 spesifikasi peluncuran yang menimpa templat peluncuran memiliki 4 tipe instans berbeda dengan prioritas yang ditetapkan, dan 3 subnet berbeda, masing–masing di Zona Ketersediaan yang terpisah. Kapasitas target untuk armada adalah 20 instance, dan opsi pembelian default adalah spot, yang mengakibatkan armada mencoba meluncurkan 20 Instans Spot dari salah satu dari 12 kumpulan kapasitas Spot berdasarkan strategi capacity-optimized-prioritized alokasi, yang menerapkan prioritas berdasarkan upaya terbaik, tetapi mengoptimalkan kapasitas terlebih dahulu.

```
{
    "SpotOptions": {
```

```
        "AllocationStrategy": "capacity-optimized-prioritized"
    },
    "LaunchTemplateConfigs": [
      {
        "LaunchTemplateSpecification":{
           "LaunchTemplateName":"ec2-fleet-lt1",
           "Version":"$Latest"
        },
        "Overrides":[
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-fae8c380",
              "Priority": 1.0
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-e7188bab",
              "Priority": 1.0
           },
           {
              "InstanceType":"c5.large",
              "SubnetId":"subnet-49e41922",
              "Priority": 1.0
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-fae8c380",
              "Priority": 2.0
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-e7188bab",
              "Priority": 2.0
           },
           {
              "InstanceType":"c5d.large",
              "SubnetId":"subnet-49e41922",
              "Priority": 2.0
           },
           {
              "InstanceType":"m5.large",
              "SubnetId":"subnet-fae8c380",
              "Priority": 3.0
           },
```

```
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 3.0
            },
            {
               "InstanceType":"m5.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 3.0
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-fae8c380",
               "Priority": 4.0
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-e7188bab",
               "Priority": 4.0
            },
            {
               "InstanceType":"m5d.large",
               "SubnetId":"subnet-49e41922",
               "Priority": 4.0
            }
         ]
      }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

Contoh 12: Menentukan parameter Systems Manager, bukan ID AMI

Contoh berikut menggunakan template peluncuran untuk menentukan konfigurasi untuk instance di armada. Dalam contoh ini, untuk`ImageId`, alih-alih menentukan ID AMI, AMI direferensikan dengan parameter Manajer Sistem. Pada peluncuran instance, parameter Systems Manager akan diselesaikan ke ID AMI.

Dalam contoh ini, parameter Systems Manager ditentukan dalam format yang valid:`resolve:ssm:golden-ami`. Ada format lain yang valid untuk parameter Systems Manager. Untuk informasi selengkapnya, lihat Gunakan parameter Systems Manager alih-alih ID AMI.

> **Note**
>
> Tipe armada harus bertipe `instant`. Jenis armada lainnya tidak mendukung penetapan parameter Manajer Sistem, bukan ID AMI.

```
{
    "LaunchTemplateData": {
        "ImageId": "resolve:ssm:golden-ami",
        "InstanceType": "m5.4xlarge",
        "TagSpecifications": [{
            "ResourceType": "instance",
            "Tags": [{
                "Key": "Name",
                "Value": "webserver"
            }]
        }]
    }
}
```

## Tetapkan batas pengeluaran untuk Armada EC2 atau Spot Fleet Anda

Anda dapat menetapkan batas berapa banyak yang bersedia Anda belanjakan per jam untuk Armada EC2 atau Armada Spot Anda. Ketika batas pengeluaran Anda tercapai, armada berhenti meluncurkan instans, meskipun kapasitas target belum tercapai.

Ada batas pengeluaran terpisah untuk On-Demand Instans dan Instans Spot.

Untuk mengonfigurasi batas pengeluaran untuk On-Demand Instans dan Instans Spot di Armada EC2 Anda

Gunakan perintah create-fleet dan parameter berikut:

- Untuk On-Demand Instans: Dalam `OnDemandOptions` struktur, tentukan batas pengeluaran Anda di `MaxTotalPrice` bidang.

- Untuk Instans Spot: Dalam `SpotOptions` struktur, tentukan batas pengeluaran Anda di `MaxTotalPrice` bidang.

Untuk mengonfigurasi batas pengeluaran untuk On-Demand Instans dan Instans Spot di Armada Spot Anda

Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI untuk mengonfigurasi batas pengeluaran Anda.

(Konsol) Saat membuat Spot Fleet, pilih kotak centang Tetapkan biaya maksimum untuk Instans Spot, lalu masukkan nilai untuk Tet apkan biaya maksimum Anda (per jam). Untuk informasi lebih lanjut, lihat langkah 6.e. diMembuat permintaan Spot Fleet menggunakan parameter yang ditentukan.

(AWS CLI) Gunakan perintah request-spot-fleet dan parameter berikut:

- Untuk On-Demand Instans: Tentukan batas pengeluaran Anda di `OnDemandMaxTotalPrice` bidang.
- Untuk Instans Spot: Tentukan batas pengeluaran Anda di `SpotMaxTotalPrice` bidang.

## Contoh

Contoh berikut menunjukkan dua skenario berbeda. Dalam contoh pertama, armada berhenti meluncurkan On-Demand Instans ketika telah memenuhi kapasitas target yang ditetapkan untuk On-Demand Instances (`OnDemandTargetCapacity`). Dalam contoh kedua, armada berhenti meluncurkan On-Demand Instans ketika telah mencapai jumlah maksimum yang bersedia Anda bayar per jam untuk On-Demand Instans (`MaxTotalPrice`).

Contoh: Menghentikan peluncur On-Demand an Instans saat kapasitas target tercapai

Diberikan permintaan untuk `m4.large` On-Demand Instans, di mana:

- On-Demand Harga: $0,10 per jam
- `OnDemandTargetCapacity`: 10
- `MaxTotalPrice`: 1,50 USD

Armada meluncurkan 10 On-Demand Instans karena total $1,00 (10 instans x $0,10) tidak melebihi $1,50 untuk Inst `MaxTotalPrice` ans. On-Demand

Contoh: Hentikan peluncur On-Demand an Instans saat harga total maksimum tercapai

Diberikan permintaan untuk `m4.large` On-Demand Instans, di mana:

- On-Demand Harga: $0,10 per jam
- `OnDemandTargetCapacity`: 10
- `MaxTotalPrice`: 0,80 USD

Jika armada meluncurkan kapasitas On-Demand target (10 On-Demand Instans), total biaya per jam akan menjadi $1,00. Ini lebih dari jumlah ($0.80) yang ditentukan `MaxTotalPrice` untuk On-Demand Instans. Untuk mencegah pengeluaran lebih dari yang bersedia Anda bayar, armada hanya meluncurkan 8 On-Demand Instans (di bawah kapasitas On-Demand target) karena meluncurkan lebih banyak akan melebihi `MaxTotalPrice` untuk On-Demand Instans.

## Instance kinerja yang dapat melonjak

Jika Anda meluncurkan Instans Spot menggunakan tipe instans performa yang dapat melonjak, serta jika Anda berencana untuk segera menggunakan Instans Spot performa yang dapat melonjak dan untuk durasi yang singkat, tanpa waktu idle untuk memperoleh kredit CPU, kami menyarankan Anda untuk meluncurkannya dalam mode Standar guna menghindari pembayaran biaya yang lebih tinggi. Jika Anda meluncurkan Instans Spot performa yang dapat melonjak dalam Mode tak terbatas dan langsung melonjakkan CPU, Anda akan menghabiskan kredit surplus untuk lonjakan. Jika Anda menggunakan instans untuk durasi yang singkat, instans tersebut tidak memiliki waktu untuk mengkumulasi kredit CPU untuk membayar kredit surplus, dan Anda akan dikenai biaya untuk kredit surplus saat Anda mengakhiri instans.

Mode tidak terbatas cocok untuk Instans Spot dengan performa yang dapat melonjak hanya jika instans tersebut berjalan cukup lama untuk mengakumlasi kredit CPU untuk lonjakan. Jika tidak, pembayaran kredit surplus membuat Instans Spot performa yang dapat melonjak lebih mahal daripada menggunakan instans lain. Untuk informasi selengkapnya, lihat Kapan menggunakan mode tak terbatas versus CPU tetap.

Kredit peluncuran dimaksudkan untuk memberikan pengalaman peluncuran awal yang produktif bagi instans T2 dengan menyediakan sumber daya komputasi yang memadai untuk mengonfigurasi instans. Peluncuran berulang dari instans T2 untuk mengakses kredit peluncuran baru tidak diizinkan. Jika Anda memerlukan CPU berkelanjutan, Anda dapat memperoleh kredit (dengan berhenti selama beberapa periode), menggunakan mode Tak Terbatas untuk Instans Spot T2, atau menggunakan tipe instans dengan CPU khusus.

# Tentukan atribut untuk pemilihan tipe instans untuk EC2 Fleet atau Spot Fleet

Saat membuat Armada EC2 atau Spot Fleet, Anda harus menentukan satu atau beberapa jenis instans untuk mengonfigurasi Instans dan On-Demand Instans Spot dalam armada. Sebagai alternatif untuk menentukan tipe instans secara manual, Anda dapat menentukan atribut yang harus dimiliki instans, dan Amazon EC2 akan mengidentifikasi semua tipe instans dengan atribut tersebut. Hal ini dikenal sebagai pemilihan tipe instans berbasis atribut. Misalnya, Anda dapat menentukan jumlah minimum dan maksimum vCPU yang diperlukan untuk instans Anda, dan armada akan meluncurkan instans menggunakan jenis instans yang tersedia yang memenuhi persyaratan vCPU tersebut.

Attribute-based pemilihan tipe instance sangat ideal untuk beban kerja dan kerangka kerja yang dapat fleksibel tentang jenis instans apa yang mereka gunakan, seperti saat menjalankan wadah atau flet web, memproses data besar, dan menerapkan integrasi berkelanjutan dan perkakas penerapan ()CI/CD.

Keuntungan

Attribute-based pemilihan jenis instance memiliki manfaat sebagai berikut:

- Gunakan jenis instans yang tepat dengan mudah — Dengan begitu banyak jenis instans yang tersedia, menemukan jenis instans yang tepat untuk beban kerja Anda dapat memakan waktu. Saat Anda menentukan atribut instans, tipe instans akan secara otomatis memiliki atribut yang diperlukan untuk beban kerja Anda.
- Konfigurasi yang disederhanakan — Untuk menentukan beberapa jenis instans secara manual untuk armada, Anda harus membuat penggantian template peluncuran terpisah untuk setiap jenis instans. Namun, dengan pemilihan tipe instans berbasis atribut, untuk menyediakan banyak tipe instans, Anda hanya perlu menentukan atribut instans dalam templat peluncuran atau dalam penimpaan templat peluncuran.
- Penggunaan otomatis jenis instans baru — Saat Anda menentukan atribut instans daripada tipe instans, armada Anda dapat menggunakan jenis instans generasi yang lebih baru saat dirilis, “pemeriksaan masa depan” konfigurasi armada.
- Fleksibilitas tipe instans — Saat Anda menentukan atribut instans daripada tipe instans, armada dapat memilih dari berbagai jenis instans untuk meluncurkan Instans Spot, yang mematuhi praktik terbaik Spot tentang fleksibilitas jenis instans.

Topik

- Cara kerja pemilihan tipe instans berbasis atribut
- Perlindungan harga
- Perlindungan kinerja
- Pertimbangan-pertimbangan
- Membuat Armada EC2 dengan pemilihan tipe instans berbasis atribut
- Buat Armada Spot dengan pemilihan tipe instans berbasis atribut
- Contoh konfigurasi Armada EC2 yang valid dan tidak valid
- Contoh konfigurasi Spot Fleet yang valid dan tidak valid
- Melihat pratinjau tipe instans dengan atribut tertentu


## Cara kerja pemilihan tipe instans berbasis atribut

Untuk menggunakan pemilihan tipe instans berbasis atribut dalam konfigurasi armada, Anda mengganti daftar tipe instans dengan daftar atribut instans yang dibutuhkan oleh instans Anda. EC2 Fleet atau Spot Fleet akan meluncurkan instans pada semua jenis instans yang tersedia yang memiliki atribut instans yang ditentukan.

Topik


- Tipe atribut instans
- Tempat mengonfigurasi pemilihan tipe instans berbasis atribut
- Cara EC2 Fleet atau Spot Fleet menggunakan pemilihan jenis instans berbasis atribut saat menyediakan armada


### Tipe atribut instans

Ada beberapa atribut instance yang dapat Anda tentukan untuk mengekspresikan persyaratan komputasi Anda, seperti:

- Jumlah vCPU — Jumlah minimum dan maksimum vCPU per instans.
- Mem ori — Minimum dan GiBs maksimum memori per instance.
- Penyimpanan lokal — Apakah menggunakan EBS atau volume penyimpanan instans untuk penyimpanan lokal.
- Kinerja burstable — Apakah akan menggunakan keluarga instance T, termasuk tipe T4g, T3a, T3, dan T2.

Untuk deskripsi setiap atribut dan nilai default, lihat InstanceRequirements di Refer ensi API Amazon EC2.

Tempat mengonfigurasi pemilihan tipe instans berbasis atribut

Bergantung pada apakah Anda menggunakan konsol atau AWS CLI, Anda dapat menentukan atribut instance untuk pemilihan jenis instans berbasis atribut sebagai berikut:

Di konsol, Anda dapat menentukan atribut instance dalam komponen konfigurasi armada berikut:

- Dalam templat peluncuran, lalu referensikan templat peluncuran dalam permintaan armada
- (Hanya Spot Fleet) Dalam permintaan armada

Di dalam AWS CLI, Anda dapat menentukan atribut instance di satu atau semua komponen konfigurasi armada berikut:

- Dalam templat peluncuran, lalu referensikan templat peluncuran dalam permintaan armada
- Dalam penimpaan templat peluncuran

  Jika Anda menginginkan campuran instans yang menggunakan AMI yang berbeda, Anda dapat menentukan atribut instans dalam banyak penimpaan templat peluncuran. Misalnya, jenis instance yang berbeda dapat menggunakan x86 dan Arm-based prosesor.
- (Hanya Spot Fleet) Dalam spesifikasi peluncuran

Cara EC2 Fleet atau Spot Fleet menggunakan pemilihan jenis instans berbasis atribut saat menyediakan armada

EC2 Fleet atau Spot Fleet menyediakan armada dengan cara berikut:

- Ini mengidentifikasi jenis instance yang memiliki atribut yang ditentukan.
- Ini menggunakan perlindungan harga untuk menentukan jenis instans mana yang akan dikecualikan.
- Ini menentukan kumpulan kapasitas dari mana ia akan mempertimbangkan untuk meluncurkan instans berdasarkan Wil AWS ayah atau Zona Ketersediaan yang memiliki jenis instans yang cocok.
- Ini menerapkan strategi alokasi yang ditentukan untuk menentukan dari kumpulan kapasitas mana untuk meluncurkan instans.

Perhatikan bahwa pemilihan jenis instans berbasis atribut tidak memilih kumpulan kapasitas untuk menyediakan armada; itulah tugas strategi ??? alokasi.

Jika Anda menentukan strategi alokasi, armada akan meluncurkan instance sesuai dengan strategi alokasi yang ditentukan.

- Untuk Instans Spot, pemilihan jenis instans berbasis atribut mendukung strategi alokasi harga yang dioptimalkan, kapasitas yang dioptimalkan, dan harga terendah. Perhatikan bahwa kami tidak merekomendasikan strategi alokasi Spot harga terendah karena strategi ini memiliki risiko gangguan tertinggi untuk Instans Spot Anda.
- Untuk On-Demand Instans, pemilihan jenis instans berbasis atribut mendukung strategi alokasi harga terendah.

- Jika tidak ada kapasitas untuk tipe instans dengan atribut instans yang ditentukan, tidak ada instans yang dapat diluncurkan, dan armada akan mengembalikan kesalahan.

## Perlindungan harga

Perlindungan harga adalah fitur yang mencegah Armada EC2 atau Armada Spot menggunakan jenis instans yang menurut Anda terlalu mahal meskipun sesuai dengan atribut yang Anda tentukan. Untuk menggunakan perlindungan harga, Anda menetapkan ambang harga. Kemudian, ketika Amazon EC2 memilih jenis instans dengan atribut Anda, itu tidak termasuk jenis instans yang dihargai di atas ambang batas Anda.

Cara Amazon EC2 menghitung ambang harga adalah sebagai berikut:

- Amazon EC2 pertama-tama mengidentifikasi jenis instans dengan harga terendah dari yang cocok dengan atribut Anda.
- Amazon EC2 kemudian mengambil nilai (dinyatakan sebagai persentase) yang Anda tentukan untuk parameter perlindungan harga dan mengalikannya dengan harga jenis instans yang diidentifikasi. Hasilnya adalah harga yang digunakan sebagai ambang harga.

Ada ambang harga terpisah untuk On-Demand Instans dan Instans Spot.

Saat Anda membuat armada dengan pemilihan jenis instans berbasis atribut, perlindungan harga diaktifkan secara default. Anda dapat menyimpan nilai default, atau Anda dapat menentukan sendiri.

Anda juga dapat mematikan perlindungan harga. Untuk menunjukkan tidak ada ambang perlindungan harga, tentukan nilai persentase tinggi, seperti999999.

Topik


- Cara mengidentifikasi jenis instans dengan harga terendah
- On-Demand Perlindungan harga instans
- Perlindungan harga Instance Spot
- Tentukan ambang perlindungan harga


## Cara mengidentifikasi jenis instans dengan harga terendah

Amazon EC2 menentukan harga yang menjadi dasar ambang harga dengan mengidentifikasi jenis instans dengan harga terendah dari yang cocok dengan atribut yang Anda tentukan. Ia melakukan ini dengan cara berikut:

- Ini pertama melihat jenis instance C, M, atau R generasi saat ini yang cocok dengan atribut Anda. Jika menemukan kecocokan, ia mengidentifikasi jenis instance dengan harga terendah.
- Jika tidak ada kecocokan, maka akan melihat jenis instance generasi saat ini yang cocok dengan atribut Anda. Jika menemukan kecocokan, ia mengidentifikasi jenis instance dengan harga terendah.
- Jika tidak ada kecocokan, maka akan melihat jenis instans generasi sebelumnya yang cocok dengan atribut Anda, dan mengidentifikasi jenis instans dengan harga terendah.

## On-Demand Perlindungan harga instans

Ambang batas perlindungan harga untuk jenis On-Demand instans dihitung sebagai persentase yang lebih tinggi dari jenis On-Demand instans dengan harga terendah yang diidentifikasi (`OnDemandMaxPricePercentageOverLowestPrice`). Anda menentukan persentase yang lebih tinggi yang bersedia Anda bayar. Jika Anda tidak menentukan parameter ini, maka nilai default `20` digunakan untuk menghitung ambang perlindungan harga 20% lebih tinggi dari harga yang diidentifikasi.

Misalnya, jika harga On-Demand instans yang diidentifikasi adalah`0.4271`, dan Anda tent `25` ukan, maka ambang harga 25% lebih tinggi dari`0.4271`. Hal ini dihitung sebagai berikut:`0.4271 * 1.25 = 0.533875`. Harga yang dihitung adalah maksimum yang bersedia Anda bayar untuk On-Demand Instans, dan, dalam contoh ini, Amazon EC2 akan mengecualikan semua jenis On-Demand instans yang harganya lebih `0.533875` mahal dari.

### Perlindungan harga Instance Spot

Secara default, Amazon EC2 akan secara otomatis menerapkan perlindungan harga Instans Spot yang optimal untuk memilih secara konsisten dari berbagai jenis instans. Anda juga dapat mengatur sendiri perlindungan harga secara manual. Namun, membiarkan Amazon EC2 melakukannya untuk Anda dapat meningkatkan kemungkinan kapasitas Spot Anda terpenuhi.

Anda dapat menentukan perlindungan harga secara manual menggunakan salah satu opsi berikut. Jika Anda mengatur perlindungan harga secara manual, sebaiknya gunakan opsi pertama.

- Per sentase jenis On-Demand instans dengan harga terendah yang diidentifikasi [`MaxSpotPriceAsPercentageOfOptimalOnDemandPrice`]

  Misalnya, jika harga jenis On-Demand instans yang diidentifikasi adalah`0.4271`, dan Anda tent `60` ukan, maka ambang harga adalah 60% dari`0.4271`. Hal ini dihitung sebagai berikut:`0.4271 * 0.60 = 0.25626`. Harga yang dihitung adalah maksimum yang bersedia Anda bayar untuk Instans Spot, dan, dalam contoh ini, Amazon EC2 akan mengecualikan semua jenis instans Spot yang harganya lebih `0.25626` mahal dari.
- Per sentase lebih tinggi dari jenis instans Spot dengan harga terendah yang diidentifikasi [`SpotMaxPricePercentageOverLowestPrice`]

  Misalnya, jika harga jenis instans Spot yang diidentifikasi adalah`0.1808`, dan Anda tent `25` ukan, maka ambang harga 25% lebih tinggi dari`0.1808`. Hal ini dihitung sebagai berikut:`0.1808 * 1.25 = 0.226`. Harga yang dihitung adalah maksimum yang bersedia Anda bayar untuk Instans Spot, dan, dalam contoh ini, Amazon EC2 akan mengecualikan semua jenis instans Spot yang harganya lebih `0.266` mahal dari. Kami tidak menyarankan menggunakan parameter ini karena harga Spot dapat berfluktuasi, dan oleh karena itu ambang perlindungan harga Anda mungkin juga berfluktuasi.

#### Tentukan ambang perlindungan harga

Untuk menentukan ambang perlindungan harga menggunakan AWS CLI

Saat membuat Armada EC2 atau Spot Fleet menggunakan AWS CLI, konfigurasikan armada untuk pemilihan jenis instans berbasis atribut, lalu lakukan hal berikut:

- Untuk menentukan ambang perlindungan harga On-Demand Instans, dalam file konfigurasi JSON, dalam `InstanceRequirements` struktur,

forOnDemandMaxPricePercentageOverLowestPrice, masukkan ambang perlindungan harga sebagai persentase.

- Untuk menentukan ambang perlindungan harga Instance Spot, dalam file konfigurasi JSON, dalam InstanceRequirements struktur, tent ukan salah satu parameter berikut:
  - UntukMaxSpotPriceAsPercentageOfOptimalOnDemandPrice, masukkan ambang perlindungan harga sebagai persentase.
  - UntukSpotMaxPricePercentageOverLowestPrice, masukkan ambang perlindungan harga sebagai persentase.

Untuk informasi selengkapnya, lihat Membuat Armada EC2 dengan pemilihan tipe instans berbasis atribut atau Buat Armada Spot dengan pemilihan tipe instans berbasis atribut.

(Hanya Spot Fleet) Untuk menentukan ambang perlindungan harga menggunakan konsol

Saat membuat Spot Fleet di konsol, konfigurasikan armada untuk pemilihan jenis instans berbasis atribut, lalu lakukan hal berikut:

- Untuk menentukan ambang perlindungan harga On-Demand Instans, di bawah atribut instans tambahan, pilih perlindungan On-demand harga, pilih Tam bah atribut, lalu masukkan ambang perlindungan harga sebagai persentase.
- Untuk menentukan ambang perlindungan harga Instans Spot, atribut instans tambahan, pilih Perlindungan harga spot, pilih Tam bah atribut, pilih nilai dasar yang menjadi dasar harga, lalu masukkan ambang perlindungan harga sebagai persentase.

> **Note**
>
> Saat membuat armada, jika Anda menyetel TargetCapacityUnitType ke vcpu ataumemory-mib, ambang perlindungan harga diterapkan berdasarkan harga per vCPU atau per memori, bukan harga per instans.

## Perlindungan kinerja

Perlindungan kinerja adalah fitur yang memastikan Armada EC2 atau Spot Fleet Anda menggunakan jenis instans yang mirip atau melebihi dasar kinerja yang ditentukan. Untuk menggunakan perlindungan kinerja, Anda menentukan keluarga instance sebagai referensi dasar. Kemampuan keluarga instance yang ditentukan menetapkan tingkat kinerja terendah yang dapat diterima. Saat

Amazon EC2 memilih jenis instans untuk armada Anda, Amazon EC2 mempertimbangkan atribut yang Anda tentukan dan dasar kinerja. Jenis instans yang berada di bawah garis dasar kinerja secara otomatis dikecualikan dari pemilihan, bahkan jika mereka cocok dengan atribut yang Anda tentukan lainnya. Ini memastikan bahwa semua jenis instans yang dipilih menawarkan kinerja yang serupa atau lebih baik dari baseline yang ditetapkan oleh keluarga instans yang ditentukan. Amazon EC2 menggunakan baseline ini untuk memandu pemilihan jenis instans, tetapi tidak ada jaminan bahwa jenis instans yang dipilih akan selalu melebihi baseline untuk setiap aplikasi.

Saat ini, fitur ini hanya mendukung kinerja CPU sebagai faktor kinerja dasar. Kinerja CPU dari prosesor CPU keluarga instans tertentu berfungsi sebagai dasar kinerja, memastikan bahwa jenis instans yang dipilih serupa atau melebihi baseline ini. Keluarga instans dengan prosesor CPU yang sama menghasilkan hasil pemfilteran yang sama, bahkan jika kinerja jaringan atau disk mereka berbeda. Misalnya, menentukan salah satu `c6in` atau `c6i` sebagai referensi dasar akan menghasilkan hasil pemfilteran berbasis kinerja yang identik karena kedua keluarga instance menggunakan prosesor CPU yang sama.

Keluarga instans yang tidak didukung

Keluarga instance berikut  tidak  didukung untuk perlindungan kinerja:

- Tujuan umum:  Mac1 | Mac2 | | Mac2-m1ultra | Mac2-m2 | M1 Mac2-m2pro | M2 | T1
- Komputasi dioptimalkan: C1
- Memori dioptimalkan:  U-3tb1 | U-6tb1 | U-9tb1 | | U-12tb1 U-18tb1 | u-24tb1 | | | | U7i-12tb U7in-16tb U7in-24tb U7in-32tb
- Komputasi yang dipercepat:  G3 | G3s | P3dN | P4d | P5
- High-performance komputasi:  HPC7g

Jika Anda mengaktifkan perlindungan kinerja dengan menentukan keluarga instans yang didukung, jenis instans yang dikembalikan akan mengecualikan keluarga instans yang tidak didukung di atas.

Jika Anda menentukan keluarga instans yang tidak didukung sebagai nilai untuk kinerja dasar, API mengembalikan respons kosong untuk  GetInstanceTypesFromInstanceRequirements  dan pengecualian untuk  CreateFleet,,  RequestSpotFleet  ModifyFleet, dan  ModifySpotFleetRequest.

Contoh: Menetapkan baseline kinerja CPU

Dalam contoh berikut, persyaratan instans adalah meluncurkan dengan tipe instance yang memiliki inti CPU yang berkinerja sama dengan keluarga `c6i` instance. Ini akan menyaring jenis instans

dengan prosesor CPU yang kurang berkinerja, bahkan jika mereka memenuhi persyaratan instans tertentu lainnya seperti jumlah vCPU. Misalnya, jika atribut instans yang Anda tentukan menyertakan 4 vCPU dan memori 16 GB, jenis instans dengan atribut ini tetapi dengan kinerja CPU yang lebih rendah daripada `c6i` akan dikecualikan dari pemilihan.

```
"BaselinePerformanceFactors": {
        "Cpu": {
            "References": [
                {
                    "InstanceFamily": "c6i"
                }
            ]
        }
```

## Pertimbangan-pertimbangan

- Anda dapat menentukan jenis instans atau atribut instans di Armada EC2 atau Spot Fleet, tetapi tidak keduanya secara bersamaan.

  Saat menggunakan CLI, penimpaan templat peluncuran akan menimpa templat peluncuran. Misalnya, jika templat peluncuran berisi tipe instans dan penimpaan templat peluncuran berisi atribut instans, instans yang diidentifikasi oleh atribut instans akan menimpa tipe instans dalam templat peluncuran.
- Saat menggunakan CLI, saat Anda menentukan atribut instans sebagai penimpaan, Anda juga tidak dapat menentukan bobot atau prioritas.
- Anda dapat menentukan maksimum empat struktur `InstanceRequirements` dalam konfigurasi permintaan.

## Membuat Armada EC2 dengan pemilihan tipe instans berbasis atribut

Anda dapat mengonfigurasi Armada EC2 untuk menggunakan pemilihan jenis instans berbasis atribut. Tidak mungkin membuat Armada EC2 menggunakan konsol Amazon EC2.

Atribut untuk pemilihan tipe instans berbasis atribut ditentukan dalam struktur `InstanceRequirements`. Ketika `InstanceRequirements` disertakan dalam konfigurasi armada, `InstanceType` dan `WeightedCapacity` harus dikecualikan; mereka tidak dapat menentukan konfigurasi armada pada saat yang sama dengan atribut instance.

Dalam PowerShell contoh AWS CLI dan, atribut berikut ditentukan:

- `VCpuCount`- Minimal 2 vCPU dan maksimal 4 vCPU. Jika Anda tidak memerlukan batas maksimum, Anda dapat menghilangkan nilai maksimum.
- `MemoryMiB`- Minimal 8 GiB memori dan maksimum 16 GiB. Jika Anda tidak memerlukan batas maksimum, Anda dapat menghilangkan nilai maksimum.

Konfigurasi ini mengidentifikasi semua jenis instance dengan 2 hingga 4 vCPU dan 8 hingga 16 GiB memori. Namun, perlindungan harga dan strategi alokasi mungkin akan mengecualikan beberapa tipe instans jika Armada EC2 menyediakan armada.

AWS CLI

Untuk membuat Armada EC2 dengan pemilihan jenis instans berbasis atribut

Gunakan perintah create-fleet untuk membuat Armada EC2. Tentukan konfigurasi armada dalam file JSON.

```
aws ec2 create-fleet \
    --region us-east-1 \
    --cli-input-json file://file_name.json
```

*file_name*`.json`File contoh berikut berisi parameter yang mengonfigurasi Armada EC2 untuk menggunakan pemilihan jenis instans berbasis atribut.

```
{
    "SpotOptions": {
        "AllocationStrategy": "price-capacity-optimized"
    },
    "LaunchTemplateConfigs": [{
        "LaunchTemplateSpecification": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1"
        },
        "Overrides": [{
            "InstanceRequirements": {
                "VCpuCount": {
                    "Min": 2,
                    "Max": 4
                },
                "MemoryMiB": {
                    "Min": 8192,
                    "Max": 16384
```

```
                }
            }
        }]
    }],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

PowerShell

Untuk membuat Armada EC2 dengan pemilihan jenis instans berbasis atribut

Gunakan New-EC2Fleet cmdlet.

```
$vcpuCount = New-Object Amazon.EC2.Model.VCpuCountRangeRequest
$vcpuCount.Min = 2
$vcpuCount.Max = 4
$memoryMiB = New-Object Amazon.EC2.Model.MemoryMiBRequest
$memoryMiB.Min = 8192
$memoryMiB.Max = 16384
$instanceRequirements = New-Object Amazon.EC2.Model.InstanceRequirementsRequest
$instanceRequirements.VCpuCount = $vcpuCount
$instanceRequirements.MemoryMiB = $memoryMiB

$launchTemplateSpec = New-Object
 Amazon.EC2.Model.FleetLaunchTemplateSpecificationRequest
$launchTemplateSpec.LaunchTemplateName = "my-launch-template"
$launchTemplateSpec.Version = "1"
$override = New-Object Amazon.EC2.Model.FleetLaunchTemplateOverridesRequest
$override.InstanceRequirements = $instanceRequirements
$launchTemplateConfig = New-Object Amazon.EC2.Model.FleetLaunchTemplateConfigRequest
$launchTemplateConfig.LaunchTemplateSpecification = $launchTemplateSpec
$launchTemplateConfig.Overrides = @($override)

New-EC2Fleet `
    -SpotOptions_AllocationStrategy "price-capacity-optimized" `
    -LaunchTemplateConfig @($launchTemplateConfig) `
    -TargetCapacitySpecification_DefaultTargetCapacityType "spot" `
    -TargetCapacitySpecification_TotalTargetCapacity 20 `
    -Type "instant"
```

## Buat Armada Spot dengan pemilihan tipe instans berbasis atribut

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Anda dapat mengonfigurasi armada untuk menggunakan pemilihan jenis instans berbasis atribut.

Atribut untuk pemilihan tipe instans berbasis atribut ditentukan dalam struktur `InstanceRequirements`. Ketika `InstanceRequirements` disertakan dalam konfigurasi armada, `InstanceType` dan `WeightedCapacity` harus dikecualikan; mereka tidak dapat menentukan konfigurasi armada pada saat yang sama dengan atribut instance.

Dalam PowerShell contoh AWS CLI dan, atribut berikut ditentukan:

- `VCpuCount`- Minimal 2 vCPU dan maksimal 4 vCPU. Jika Anda tidak memerlukan batas maksimum, Anda dapat menghilangkan nilai maksimum.
- `MemoryMiB`- Minimal 8 GiB memori dan maksimum 16 GiB. Jika Anda tidak memerlukan batas maksimum, Anda dapat menghilangkan nilai maksimum.

Konfigurasi ini mengidentifikasi setiap jenis instance yang memiliki 2 hingga 4 vCPU dan 8 hingga 16 GiB memori. Namun, perlindungan harga dan strategi alokasi mungkin akan mengecualikan beberapa tipe instans jika Armada Spot menyediakan armada.

Console

Untuk mengonfigurasi Spot Fleet untuk pemilihan jenis instans berbasis atribut

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot, lalu pilih Minta Instans Spot.
3. Ikuti langkah-langkah ini untuk membuat Armada Spot. Untuk informasi selengkapnya, lihat Membuat permintaan Spot Fleet menggunakan parameter yang ditentukan.

   Saat membuat Armada Spot, konfigurasikan armada untuk pemilihan tipe instans berbasis atribut sebagai berikut:

a. Untuk Persyaratan tipe instans, pilih Tentukan atribut instans yang sesuai dengan persyaratan komputasi Anda.
b. Untuk vCPU, masukkan jumlah minimum dan maksimum vCPU yang diinginkan. Untuk menentukan tanpa batasan, pilih Tanpa minimum, Tanpa maksimum, atau keduanya.
c. Untuk Memori (GiB), masukkan jumlah memori minimum dan maksimum yang diinginkan. Untuk menentukan tanpa batasan, pilih Tanpa minimum, Tanpa maksimum, atau keduanya.
d. (Opsional) Untuk atribut instans Tambahan, Anda dapat secara opsional menentukan satu atau lebih atribut untuk mengekspresikan kebutuhan komputasi Anda secara lebih mendetail. Setiap atribut tambahan menambahkan batasan lebih lanjut untuk permintaan Anda.
e. (Opsional) Perluas Pratinjau tipe instans yang cocok untuk melihat tipe instans yang memiliki atribut yang Anda tentukan.

AWS CLI

Untuk mengonfigurasi Spot Fleet untuk pemilihan jenis instans berbasis atribut

Gunakan perintah request-spot-fleet untuk membuat Spot Fleet. Tentukan konfigurasi armada dalam file JSON.

```
aws ec2 request-spot-fleet \
    --region us-east-1 \
    --spot-fleet-request-config file://file_name.json
```

*file_name*`.json`File contoh berikut berisi parameter yang mengonfigurasi Spot Fleet untuk menggunakan pemilihan jenis instans berbasis atribut.

```
{
    "AllocationStrategy": "priceCapacityOptimized",
    "TargetCapacity": 20,
    "Type": "request",
    "LaunchTemplateConfigs": [{
        "LaunchTemplateSpecification": {
            "LaunchTemplateName": "my-launch-template",
            "Version": "1"
        },
        "Overrides": [{
```

```
            "InstanceRequirements": {
                "VCpuCount": {
                    "Min": 2,
                    "Max": 4
                },
                "MemoryMiB": {
                    "Min": 8192,
                    "Max": 16384
                }
            }
        }]
    }]
}
```

PowerShell

Untuk mengonfigurasi Spot Fleet untuk pemilihan jenis instans berbasis atribut

Gunakan Request-EC2SpotFleet cmdlet.

```
$vcpuCount = New-Object Amazon.EC2.Model.VCpuCountRange
$vcpuCount.Min = 2
$vcpuCount.Max = 4
$memoryMiB = New-Object Amazon.EC2.Model.MemoryMiB
$memoryMiB.Min = 8192
$memoryMiB.Max = 16384
$instanceRequirements = New-Object Amazon.EC2.Model.InstanceRequirements
$instanceRequirements.VCpuCount = $vcpuCount
$instanceRequirements.MemoryMiB = $memoryMiB

$launchTemplateSpec = New-Object Amazon.EC2.Model.FleetLaunchTemplateSpecification
$launchTemplateSpec.LaunchTemplateName = "my-launch-template"
$launchTemplateSpec.Version = "1"
$override = New-Object Amazon.EC2.Model.LaunchTemplateOverrides
$override.InstanceRequirements = $instanceRequirements
$launchTemplateConfig = New-Object Amazon.EC2.Model.LaunchTemplateConfig
$launchTemplateConfig.LaunchTemplateSpecification = $launchTemplateSpec
$launchTemplateConfig.Overrides = @($override)

Request-EC2SpotFleet `
    -SpotFleetRequestConfig_AllocationStrategy "PriceCapacityOptimized" `
    -SpotFleetRequestConfig_TargetCapacity 20 `
    -SpotFleetRequestConfig_Type "Request" `
```

```
    -SpotFleetRequestConfig_LaunchTemplateConfig $launchTemplateConfig
```

## Contoh konfigurasi Armada EC2 yang valid dan tidak valid

Jika Anda menggunakan AWS CLI untuk membuat Armada EC2, Anda harus memastikan bahwa konfigurasi armada Anda valid. Contoh berikut menunjukkan konfigurasi yang valid dan tidak valid.

Konfigurasi dianggap tidak valid jika berisi hal berikut:

- Struktur `Overrides` tunggal dengan `InstanceRequirements` maupun `InstanceType`
- Dua struktur `Overrides`, satu dengan `InstanceRequirements` dan yang lainnya dengan `InstanceType`
- Dua struktur `InstanceRequirements` dengan nilai atribut yang tumpang tindih dalam `LaunchTemplateSpecification` yang sama

Contoh konfigurasi

- Konfigurasi yang valid: Templat peluncuran tunggal dengan penimpaan
- Konfigurasi yang valid: Template peluncuran tunggal dengan beberapa InstanceRequirements
- Konfigurasi yang valid: Dua templat peluncuran, masing-masing dengan penimpaan
- Konfigurasi yang valid: Hanya InstanceRequirements yang ditentukan, tidak ada nilai atribut yang tumpang tindih
- Konfigurasi tidak valid: Override berisi dan InstanceRequirements InstanceType
- Konfigurasi tidak valid: Dua Override berisi dan InstanceRequirements InstanceType
- Konfigurasi tidak valid: Nilai atribut tumpang tindih

### Konfigurasi yang valid: Templat peluncuran tunggal dengan penimpaan

Konfigurasi berikut ini valid. Konfigurasi ini berisi satu templat peluncuran dan satu struktur `Overrides` yang berisi satu struktur `InstanceRequirements`. Berikut ini adalah penjelasan teks mengenai contoh konfigurasi.

```
{
        "LaunchTemplateConfigs": [
        {
```

```
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "My-launch-template",
                "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 2,
                            "Max": 8
                        },
                        "MemoryMib": {
                            "Min": 0,
                            "Max": 10240
                        },
                        "MemoryGiBPerVCpu": {
                            "Max": 10000
                        },
                        "RequireHibernateSupport": true
                    }
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 5000,
        "DefaultTargetCapacityType": "spot",
        "TargetCapacityUnitType": "vcpu"
        }
    }
}
```

### `InstanceRequirements`

Untuk menggunakan pemilihan instans berbasis atribut, Anda harus menyertakan struktur `InstanceRequirements` dalam konfigurasi armada, dan menentukan atribut yang diinginkan untuk instans tersebut di armada.

Pada contoh sebelumnya, atribut instans berikut ini ditentukan:

- `VCpuCount` – Tipe instans harus memiliki minimum 2 dan maksimum 8 vCPU.
- `MemoryMiB` – Tipe instans harus memiliki memori maksimum 10240 MiB. Minimum 0 menunjukkan bahwa tidak ada batas minimum.

- `MemoryGiBPerVCpu` – Tipe instans harus memiliki memori maksimum 10.000 GiB per vCPU. Parameter `Min` bersifat opsional. Dengan menghilangkannya, Anda mengindikasikan tidak ada batas minimum.

### `TargetCapacityUnitType`

Parameter `TargetCapacityUnitType` menentukan unit untuk kapasitas target. Dalam contoh, kapasitas targetnya adalah `5000` dan tipe unit kapasitas targetnya adalah `vcpu`, yang bersama-sama menentukan kapasitas target yang diinginkan sebesar 5.000 vCPU. Armada EC2 akan meluncurkan instans yang cukup sehingga jumlah total vCPU dalam armada adalah 5.000 vCPU.

Konfigurasi yang valid: Template peluncuran tunggal dengan beberapa InstanceRequirements

Konfigurasi berikut ini valid. Konfigurasi ini berisi satu templat peluncuran dan satu struktur `Overrides` yang berisi dua struktur `InstanceRequirements`. Atribut yang ditentukan di `InstanceRequirements` valid karena nilainya tidak tumpang tindih—`InstanceRequirements` struktur pertama menentukan `VCpuCount` 0-2 vCPU, sedangkan struktur `InstanceRequirements` kedua menentukan 4-8 vCPU.

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
```

```
                            "Min": 4,
                            "Max": 8
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
        "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

Konfigurasi yang valid: Dua templat peluncuran, masing-masing dengan penimpaan

Konfigurasi berikut ini valid. Konfigurasi ini berisi dua templat peluncuran, masing-masing dengan satu struktur `Overrides` yang berisi satu struktur `InstanceRequirements`. Konfigurasi ini berguna untuk dukungan arsitektur `arm` dan `x86` dalam armada yang sama.

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "armLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
```

```
                {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "x86LaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
         "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

Konfigurasi yang valid: Hanya **`InstanceRequirements`** yang ditentukan, tidak ada nilai atribut yang tumpang tindih

Konfigurasi berikut ini valid. Konfigurasi ini berisi dua struktur `LaunchTemplateSpecification`, masing-masing dengan templat peluncuran dan struktur `Overrides` yang berisi struktur `InstanceRequirements`. Atribut yang ditentukan di `InstanceRequirements` valid karena nilainya tidak tumpang tindih—`InstanceRequirements` struktur pertama menentukan `VCpuCount` 0-2 vCPU, sedangkan struktur `InstanceRequirements` kedua menentukan 4-8 vCPU.

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
```

```
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            },
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 4,
                            "Max": 8
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
        "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

Konfigurasi tidak valid: **`Override`** berisi dan **`InstanceRequirements InstanceType`**

Konfigurasi berikut ini tidak valid. Struktur `Overrides` berisi `InstanceRequirements` dan`InstanceType`. Untuk `Overrides`, Anda dapat menentukan antara `InstanceRequirements` atau `InstanceType`, tetapi tidak keduanya.

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
        "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

Konfigurasi tidak valid: Dua **`Override`** berisi dan **`InstanceRequirements InstanceType`**

Konfigurasi berikut ini tidak valid. Struktur `Overrides` berisi `InstanceRequirements` dan`InstanceType`. Anda dapat menentukan antara `InstanceRequirements` atau `InstanceType`, tetapi tidak keduanya, meskipun berada dalam struktur `Overrides` yang berbeda.

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            },
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
         "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
    }
}
```

Konfigurasi tidak valid: Nilai atribut tumpang tindih

Konfigurasi berikut ini tidak valid. Dua struktur `InstanceRequirements` masing-masing berisi `"VCpuCount": {"Min": 0, "Max": 2}`. Nilai untuk atribut ini tumpang tindih, yang akan mengakibatkan kolam kapasitas ganda.

```
{
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    },
                    {
                      "InstanceRequirements": {
                          "VCpuCount": {
                              "Min": 0,
                              "Max": 2
                          },
                          "MemoryMiB": {
                              "Min": 0
                          }
                      }
                  }
                }
              ]
            }
        ],
         "TargetCapacitySpecification": {
            "TotalTargetCapacity": 1,
            "DefaultTargetCapacityType": "spot"
        }
```

```
    }
}
```

## Contoh konfigurasi Spot Fleet yang valid dan tidak valid

Jika Anda menggunakan AWS CLI untuk membuat Spot Fleet, Anda harus memastikan bahwa konfigurasi armada Anda valid. Contoh berikut menunjukkan konfigurasi yang valid dan tidak valid.

Konfigurasi dianggap tidak valid jika berisi hal berikut:

- Struktur `Overrides` tunggal dengan `InstanceRequirements` maupun `InstanceType`
- Dua struktur `Overrides`, satu dengan `InstanceRequirements` dan yang lainnya dengan `InstanceType`
- Dua struktur `InstanceRequirements` dengan nilai atribut yang tumpang tindih dalam `LaunchTemplateSpecification` yang sama

Contoh konfigurasi

- Konfigurasi yang valid: Templat peluncuran tunggal dengan penimpaan
- Konfigurasi yang valid: Template peluncuran tunggal dengan beberapa InstanceRequirements
- Konfigurasi yang valid: Dua templat peluncuran, masing-masing dengan penimpaan
- Konfigurasi yang valid: Hanya InstanceRequirements yang ditentukan, tidak ada nilai atribut yang tumpang tindih
- Konfigurasi tidak valid: Override berisi dan InstanceRequirements InstanceType
- Konfigurasi tidak valid: Dua Override berisi dan InstanceRequirements InstanceType
- Konfigurasi tidak valid: Nilai atribut tumpang tindih

### Konfigurasi yang valid: Templat peluncuran tunggal dengan penimpaan

Konfigurasi berikut ini valid. Konfigurasi ini berisi satu templat peluncuran dan satu struktur `Overrides` yang berisi satu struktur `InstanceRequirements`. Berikut ini adalah penjelasan teks mengenai contoh konfigurasi.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
```

```
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "My-launch-template",
                "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 2,
                            "Max": 8
                        },
                        "MemoryMib": {
                            "Min": 0,
                            "Max": 10240
                        },
                        "MemoryGiBPerVCpu": {
                            "Max": 10000
                        },
                        "RequireHibernateSupport": true
                    }
                }
            ]
        }
    ],
        "TargetCapacity": 5000,
            "OnDemandTargetCapacity": 0,
            "TargetCapacityUnitType": "vcpu"
    }
}
```

### `InstanceRequirements`

Untuk menggunakan pemilihan instans berbasis atribut, Anda harus menyertakan struktur `InstanceRequirements` dalam konfigurasi armada, dan menentukan atribut yang diinginkan untuk instans tersebut di armada.

Pada contoh sebelumnya, atribut instans berikut ini ditentukan:

- `VCpuCount` – Tipe instans harus memiliki minimum 2 dan maksimum 8 vCPU.

- `MemoryMiB` – Tipe instans harus memiliki memori maksimum 10240 MiB. Minimum 0 menunjukkan bahwa tidak ada batas minimum.
- `MemoryGiBPerVCpu` – Tipe instans harus memiliki memori maksimum 10.000 GiB per vCPU. Parameter `Min` bersifat opsional. Dengan menghilangkannya, Anda mengindikasikan tidak ada batas minimum.

### `TargetCapacityUnitType`

Parameter `TargetCapacityUnitType` menentukan unit untuk kapasitas target. Dalam contoh, kapasitas targetnya adalah `5000` dan tipe unit kapasitas targetnya adalah `vcpu`, yang bersama-sama menentukan kapasitas target yang diinginkan sebesar 5.000 vCPU. Armada Spot akan meluncurkan instans yang cukup sehingga jumlah total vCPU dalam armada adalah 5.000 vCPU.

Konfigurasi yang valid: Template peluncuran tunggal dengan beberapa InstanceRequirements

Konfigurasi berikut ini valid. Konfigurasi ini berisi satu templat peluncuran dan satu struktur `Overrides` yang berisi dua struktur `InstanceRequirements`. Atribut yang ditentukan di `InstanceRequirements` valid karena nilainya tidak tumpang tindih—`InstanceRequirements` struktur pertama menentukan `VCpuCount` 0-2 vCPU, sedangkan struktur `InstanceRequirements` kedua menentukan 4-8 vCPU.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
```

```
                        "Min": 0
                    }
                }
            },
            {
                "InstanceRequirements": {
                    "VCpuCount": {
                        "Min": 4,
                        "Max": 8
                    },
                    "MemoryMiB": {
                        "Min": 0
                    }
                }
            }
          ]
        }
    ],
    "TargetCapacity": 1,
    "OnDemandTargetCapacity": 0,
    "Type": "maintain"
  }
}
```

Konfigurasi yang valid: Dua templat peluncuran, masing-masing dengan penimpaan

Konfigurasi berikut ini valid. Konfigurasi ini berisi dua templat peluncuran, masing-masing dengan satu struktur `Overrides` yang berisi satu struktur `InstanceRequirements`. Konfigurasi ini berguna untuk dukungan arsitektur `arm` dan `x86` dalam armada yang sama.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "armLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
```

```
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "x86LaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

Konfigurasi yang valid: Hanya **`InstanceRequirements`** yang ditentukan, tidak ada nilai atribut yang tumpang tindih

Konfigurasi berikut ini valid. Konfigurasi ini berisi dua struktur `LaunchTemplateSpecification`, masing-masing dengan templat peluncuran dan struktur `Overrides` yang berisi struktur `InstanceRequirements`. Atribut yang ditentukan di `InstanceRequirements` valid karena

nilainya tidak tumpang tindih—`InstanceRequirements` struktur pertama menentukan `VCpuCount` 0-2 vCPU, sedangkan struktur `InstanceRequirements` kedua menentukan 4-8 vCPU.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            },
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 4,
                            "Max": 8
                        },
                        "MemoryMiB": {
                            "Min": 0
```

```
                        }
                    }
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

Konfigurasi tidak valid: **`Override`** berisi dan **`InstanceRequirements InstanceType`**

Konfigurasi berikut ini tidak valid. Struktur `Overrides` berisi `InstanceRequirements` dan`InstanceType`. Untuk `Overrides`, Anda dapat menentukan antara `InstanceRequirements` atau `InstanceType`, tetapi tidak keduanya.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                },
                {
```

```
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

Konfigurasi tidak valid: Dua **`Override`** berisi dan **`InstanceRequirements InstanceType`**

Konfigurasi berikut ini tidak valid. Struktur `Overrides` berisi `InstanceRequirements` dan`InstanceType`. Anda dapat menentukan antara `InstanceRequirements` atau `InstanceType`, tetapi tidak keduanya, meskipun berada dalam struktur `Overrides` yang berbeda.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
                        "MemoryMiB": {
                            "Min": 0
                        }
                    }
                }
              ]
            },
```

```
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyOtherLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceType": "m5.large"
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

Konfigurasi tidak valid: Nilai atribut tumpang tindih

Konfigurasi berikut ini tidak valid. Dua struktur `InstanceRequirements` masing-masing berisi `"VCpuCount": {"Min": 0, "Max": 2}`. Nilai untuk atribut ini tumpang tindih, yang akan mengakibatkan kolam kapasitas ganda.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "MyLaunchTemplate",
                    "Version": "1"
                },
                "Overrides": [
                {
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 2
                        },
```

```
                        "MemoryMiB": {
                            "Min": 0
                        }
                    },
                    {
                      "InstanceRequirements": {
                          "VCpuCount": {
                              "Min": 0,
                              "Max": 2
                          },
                          "MemoryMiB": {
                              "Min": 0
                          }
                      }
                  }
                }
              ]
            }
        ],
        "TargetCapacity": 1,
        "OnDemandTargetCapacity": 0,
        "Type": "maintain"
    }
}
```

## Melihat pratinjau tipe instans dengan atribut tertentu

Anda dapat menggunakan perintah get-instance-types-from-instance-requirements untuk melihat pratinjau jenis instance yang cocok dengan atribut yang Anda tentukan. Hal ini sangat berguna untuk mengetahui atribut yang akan ditentukan dalam konfigurasi permintaan Anda tanpa meluncurkan instans apa pun. Perhatikan bahwa perintah tidak mempertimbangkan kapasitas yang tersedia.

Untuk melihat daftar jenis instance dengan menentukan atribut menggunakan AWS CLI

1. (Opsional) Untuk menghasilkan semua kemungkinan atribut yang dapat ditentukan, gunakan perintah get-instance-types-from-instance-requirements dan parameter `--generate-cli-skeleton`. Anda dapat secara opsional mengarahkan output ke file untuk menyimpannya dengan menggunakan `input > attributes.json`.

   ```
   aws ec2 get-instance-types-from-instance-requirements \
       --region us-east-1 \
   ```

```
    --generate-cli-skeleton input > attributes.json
```

Output yang diharapkan

```
{
    "DryRun": true,
    "ArchitectureTypes": [
        "i386"
    ],
    "VirtualizationTypes": [
        "hvm"
    ],
    "InstanceRequirements": {
        "VCpuCount": {
            "Min": 0,
            "Max": 0
        },
        "MemoryMiB": {
            "Min": 0,
            "Max": 0
        },
        "CpuManufacturers": [
            "intel"
        ],
        "MemoryGiBPerVCpu": {
            "Min": 0.0,
            "Max": 0.0
        },
        "ExcludedInstanceTypes": [
            ""
        ],
        "InstanceGenerations": [
            "current"
        ],
        "SpotMaxPricePercentageOverLowestPrice": 0,
        "OnDemandMaxPricePercentageOverLowestPrice": 0,
        "BareMetal": "included",
        "BurstablePerformance": "included",
        "RequireHibernateSupport": true,
        "NetworkInterfaceCount": {
            "Min": 0,
            "Max": 0
        },
```

```
        "LocalStorage": "included",
        "LocalStorageTypes": [
            "hdd"
        ],
        "TotalLocalStorageGB": {
            "Min": 0.0,
            "Max": 0.0
        },
        "BaselineEbsBandwidthMbps": {
            "Min": 0,
            "Max": 0
        },
        "AcceleratorTypes": [
            "gpu"
        ],
        "AcceleratorCount": {
            "Min": 0,
            "Max": 0
        },
        "AcceleratorManufacturers": [
            "nvidia"
        ],
        "AcceleratorNames": [
            "a100"
        ],
        "AcceleratorTotalMemoryMiB": {
            "Min": 0,
            "Max": 0
        },
        "NetworkBandwidthGbps": {
            "Min": 0.0,
            "Max": 0.0
        },
        "AllowedInstanceTypes": [
            ""
        ]
    },
    "MaxResults": 0,
    "NextToken": ""
}
```

2. Buat file konfigurasi JSON menggunakan output dari langkah sebelumnya, dan konfigurasikan sebagai berikut:

> **Note**
>
> Anda harus memberikan nilai untuk `ArchitectureTypes`, `VirtualizationTypes`, `VCpuCount`, dan `MemoryMiB`. Anda dapat menghilangkan atribut lainnya; saat dihilangkan, nilai default digunakan.
> Untuk deskripsi setiap atribut dan nilai defaultnya, lihat get- instance-types-from-instance-requirements.

a. Untuk `ArchitectureTypes`, tentukan satu atau lebih tipe arsitektur prosesor.

b. Untuk `VirtualizationTypes`, tentukan satu atau lebih tipe virtualisasi.

c. Untuk `VCpuCount`, tentukan jumlah minimum dan maksimum vCPU. Untuk menentukan tidak ada batas minimum, untuk `Min`, tentukan `0`. Untuk menentukan tidak ada batas maksimum, hilangkan parameter `Max`.

d. Untuk `MemoryMiB`, tentukan jumlah memori minimum dan maksimum dalam MiB. Untuk menentukan tidak ada batas minimum, untuk `Min`, tentukan `0`. Untuk menentukan tidak ada batas maksimum, hilangkan parameter `Max`.

e. Anda dapat secara opsional menentukan satu atau lebih atribut lainnya untuk lebih membatasi daftar tipe instans yang dikembalikan.

3. Untuk melihat pratinjau tipe instans yang memiliki atribut yang Anda tentukan dalam file JSON, gunakan perintah get-instance-types-from-instance-requirements, dan tentukan nama serta jalur ke file JSON Anda dengan menggunakan parameter `--cli-input-json`. Anda dapat secara opsional memformat output untuk muncul dalam format tabel.

```
aws ec2 get-instance-types-from-instance-requirements \
    --cli-input-json file://attributes.json \
    --output table
```

Contoh file *attributes.json*

Dalam contoh ini, atribut yang diperlukan disertakan dalam file JSON. Atribut tersebut adalah `ArchitectureTypes`, `VirtualizationTypes`, `VCpuCount`, dan `MemoryMiB`. Selain itu, atribut `InstanceGenerations` opsional juga disertakan. Perhatikan bahwa untuk `MemoryMiB`, nilai `Max` dapat dihilangkan untuk menunjukkan bahwa tidak ada batasan.

```
{
```

```
    "ArchitectureTypes": [
        "x86_64"
    ],
    "VirtualizationTypes": [
        "hvm"
    ],
    "InstanceRequirements": {
        "VCpuCount": {
            "Min": 4,
            "Max": 6
        },
        "MemoryMiB": {
            "Min": 2048
        },
        "InstanceGenerations": [
            "current"
        ]
    }
}
```

Contoh output

```
-----------------------------------------
|GetInstanceTypesFromInstanceRequirements|
+---------------------------------------+
||             InstanceTypes             ||
|+-------------------------------------+|
||             InstanceType              ||
|+-------------------------------------+|
||  c4.xlarge                          ||
||  c5.xlarge                          ||
||  c5a.xlarge                         ||
||  c5ad.xlarge                        ||
||  c5d.xlarge                         ||
||  c5n.xlarge                         ||
||  d2.xlarge                          ||
...
```

4. Setelah mengidentifikasi tipe instans yang memenuhi kebutuhan Anda, catatlah atribut instans yang Anda gunakan sehingga Anda dapat menggunakannya saat mengonfigurasi permintaan armada.

# Gunakan pembobotan instans untuk mengelola biaya dan kinerja Armada EC2 atau Spot Fleet

Dengan bobot instans, Anda menetapkan bobot untuk setiap jenis instans di Armada EC2 atau Armada Spot untuk mewakili kapasitas komputasi dan kinerja relatif satu sama lain. Berdasarkan bobot, armada dapat menggunakan kombinasi apa pun dari jenis instans yang ditentukan, selama dapat memenuhi kapasitas target yang diinginkan. Ini dapat membantu Anda mengelola biaya dan kinerja armada Anda.

Bobot mewakili unit kapasitas yang dikontribusikan oleh jenis instans terhadap kapasitas target total.

Contoh: Menggunakan bobot instans untuk manajemen kinerja

Misalkan armada Anda memiliki dua jenis instans, dan Anda menetapkan bobot yang berbeda untuk setiap jenis instans untuk mencerminkan berapa banyak yang Anda butuhkan untuk mencapai kinerja yang sama, sebagai berikut:

- `m5.large`— berat: 1
- `m5.2xlarge`— berat: 4

Dengan menetapkan bobot ini, Anda mengatakan bahwa Anda memerlukan 4 `m5.large` instance untuk mencapai kinerja yang sama dengan 1`m5.2xlarge`.

Untuk menghitung berapa banyak instance dari setiap jenis instans yang diperlukan untuk kapasitas target tertentu, gunakan rumus berikut:

```
target capacity / weight = number of instances
```

Jika kapasitas target Anda adalah 8 unit, armada dapat memenuhi kapasitas target dengan salah `m5.large` satu ataum`5.2xlarge`, atau campuran keduanya, sebagai berikut:

- 8 `m5.large` contoh (kapasitas 8/berat 1 = 8 instans)
- 2 `m5.2xlarge` contoh (kapasitas 8/berat 4 = 2 contoh)
- 4 `m5.large` dan 1 `m5.2xlarge`

Contoh: Menggunakan pembobotan instans untuk manajemen biaya

Secara default, harga yang Anda tentukan adalah per jam instans. Saat Anda menggunakan fitur pembobotan instans, harga yang Anda tentukan adalah per unit jam. Anda dapat menghitung harga

per unit jam dengan membagi harga tipe instans dengan jumlah unit yang diwakilinya. Armada menghitung jumlah instans yang akan diluncurkan dengan membagi kapasitas target dengan bobot instans. Jika hasilnya bukan bilangan bulat, armada akan membulatkannya ke bilangan bulat berikutnya, sehingga ukuran armada Anda tidak berada di bawah kapasitas targetnya. Armada dapat memilih kolam mana pun yang Anda tentukan dalam spesifikasi peluncuran, meskipun kapasitas instans yang diluncurkan melebihi kapasitas target yang diminta.

Tabel berikut mencakup contoh perhitungan untuk menentukan harga per unit untuk armada dengan kapasitas target 10.

| Tipe instans | Bobot instans | Kapasitas target | Jumlah instans yang diluncurkan | Harga per jam instans | Harga per unit jam |
|---|---|---|---|---|---|
| `r3.xlarge` | 2 | 10 | 5<br><br>(10 dibagi 2) | $0,05 | $0,025<br><br>(,05 dibagi 2) |
| `r3.8xlarge` | 8 | 10 | 2<br><br>(10 dibagi 8, hasil dibulatkan) | $0,10 | $0,0125<br><br>(,10 dibagi 8) |

Gunakan bobot instans armada sebagai berikut untuk menyediakan kapasitas target yang Anda inginkan di kumpulan dengan harga terendah per unit pada saat pemenuhan:

1. Tetapkan kapasitas target untuk armada Anda baik dalam instance (default) atau dalam unit pilihan Anda, seperti vCPU, memori, penyimpanan, atau throughput.
2. Tetapkan harga per unit.
3. Untuk setiap spesifikasi peluncuran, tentukan bobot, yang merupakan jumlah unit yang diwakili oleh tipe instans terhadap kapasitas target.

Contoh pembobotan instans

Pertimbangkan permintaan armada dengan konfigurasi berikut:

- Kapasitas target 24
- Spesifikasi peluncuran dengan tipe instans `r3.2xlarge` dan bobot 6
- Spesifikasi peluncuran dengan tipe instans `c3.xlarge` dan bobot 5

Bobot mewakili jumlah unit yang diwakili oleh tipe instans terhadap kapasitas target. Jika spesifikasi peluncuran pertama memberikan harga terendah per unit (harga `r3.2xlarge` per instans jam dibagi 6), armada akan meluncurkan empat dari instance ini (24 dibagi 6).

Jika spesifikasi peluncuran kedua memberikan harga terendah per unit (harga untuk `c3.xlarge` per instans jam dibagi 5), armada akan meluncurkan lima dari instance ini (24 dibagi 5, hasil dibulatkan ke atas).

Pembobotan instans dan strategi alokasi

Pertimbangkan permintaan armada dengan konfigurasi berikut:

- Kapasitas target 30 Instans Spot
- Spesifikasi peluncuran dengan tipe instans `c3.2xlarge` dan bobot 8
- Spesifikasi peluncuran dengan tipe instans `m3.xlarge` dan bobot 8
- Spesifikasi peluncuran dengan tipe instans `r3.xlarge` dan bobot 8

Armada akan meluncurkan empat contoh (30 dibagi 8, hasil dibulatkan ke atas). Dengan strategi `diversified`, armada meluncurkan satu instans di masing-masing dari ketiga kolam tersebut, dan instans keempat di kolam mana pun yang memberikan harga terendah per unit.

## Gunakan strategi alokasi untuk menentukan bagaimana EC2 Fleet atau Spot Fleet memenuhi Spot dan kapasitas On-Demand

Saat Anda menggunakan beberapa kumpulan kapasitas (masing-masing terdiri dari jenis instans dan Zona Ketersediaan) di Armada EC2 atau Armada Spot, Anda dapat menggunakan strategi alokasi untuk mengelola cara Amazon EC2 memenuhi Spot dan On-Demand kapasitas Anda dari kumpulan ini. Strategi alokasi dapat mengoptimalkan kapasitas, harga, dan jenis instans yang tersedia untuk digunakan. Ada berbagai strategi alokasi untuk Instans Spot dan On-Demand Instans.

Topik

- Strategi alokasi untuk Instans Spot
- Strategi alokasi untuk On-Demand Instans
- Pilih strategi alokasi Spot yang sesuai
- Mempertahankan kapasitas target untuk Instans Spot
- Prioritaskan jenis instans untuk On-Demand kapasitas


## Strategi alokasi untuk Instans Spot

Konfigurasi peluncuran menentukan semua kumpulan kapasitas Spot yang mungkin (jenis instans dan Zona Ketersediaan) dari mana Armada EC2 atau Spot Fleet dapat meluncurkan Instans Spot. Namun, saat meluncurkan instance, armada menggunakan strategi alokasi yang Anda tentukan untuk memilih kumpulan tertentu dari semua kumpulan yang mungkin.

> **Note**
>
> (Hanya instans Linux) Jika Anda mengonfigurasi Instans Spot untuk diluncurkan dengan AMD di SEV-SNP aktifkan, Anda akan dikenakan biaya penggunaan tambahan per jam yang setara dengan 10% dari tarif On-Demand per jam untuk jenis instans yang dipilih. Jika strategi alokasi menggunakan harga sebagai input, armada tidak termasuk biaya tambahan ini; hanya harga Spot yang digunakan.

Anda dapat menentukan salah satu strategi alokasi berikut untuk Instans Spot:

Kapasitas harga dioptimalkan (disarankan)

Armada mengidentifikasi kumpulan dengan ketersediaan kapasitas tertinggi untuk jumlah instans yang diluncurkan. Ini berarti bahwa Instans Spot diminta dari kumpulan yang kemungkinan besar memiliki kemungkinan terganggu terendah dalam waktu dekat. Armada kemudian meminta Instans Spot dari kumpulan dengan harga terendah.

Strategi alokasi kapasitas harga yang dioptimalkan adalah pilihan terbaik untuk sebagian besar beban kerja Spot, seperti aplikasi kontainer tanpa status, layanan mikro, aplikasi web, pekerjaan data dan analitik, dan pemrosesan batch.

Jika Anda menggunakan AWS CLI, nama parameternya adalah `price-capacity-optimized` untuk Armada EC2 dan `priceCapacityOptimized` untuk Armada Spot.

Kapasitas dioptimalkan

Armada mengidentifikasi kumpulan dengan ketersediaan kapasitas tertinggi untuk jumlah instans yang diluncurkan. Ini berarti bahwa Instans Spot diminta dari kumpulan yang kemungkinan besar memiliki kemungkinan terganggu terendah dalam waktu dekat.

Dengan Instans Spot, harga berubah secara perlahan dari waktu ke waktu berdasarkan tren penawaran dan permintaan jangka panjang, tetapi kapasitas berfluktuasi secara waktu nyata. Strategi kapasitas yang dioptimalkan secara otomatis meluncurkan Instans Spot ke kumpulan yang paling tersedia dengan melihat data kapasitas real-time dan memprediksi mana yang paling tersedia. Ini berfungsi dengan baik untuk beban kerja yang mungkin memiliki biaya interupsi yang lebih tinggi terkait dengan memulai kembali tugas, seperti Integrasi Berkelanjutan (CI), rendering gambar dan media, beban kerja Deep Learning, dan Komputasi Performa Tinggi (HPC) yang mungkin memiliki biaya interupsi yang lebih tinggi terkait dengan memulai ulang pekerjaan. Dengan menawarkan kemungkinan gangguan yang lebih sedikit, strategi kapasitas yang dioptimalkan dapat menurunkan biaya keseluruhan beban kerja Anda.

Atau, Anda dapat menggunakan strategi alokasi prioritas yang dioptimalkan kapasitas dengan parameter prioritas untuk mengurutkan jenis instans dari prioritas tertinggi ke terendah. Anda dapat mengatur prioritas yang sama untuk tipe instans yang berbeda. Armada akan mengoptimalkan kapasitas terlebih dahulu, tetapi akan menghormati prioritas jenis instans berdasarkan upaya terbaik (misalnya, jika menghormati prioritas tidak akan secara signifikan mempengaruhi kemampuan armada untuk menyediakan kapasitas optimal). Ini adalah pilihan opsi yang bagus untuk beban kerja di mana kemungkinan gangguan harus diminimalkan dan preferensi untuk tipe instans tertentu menjadi penting. Perhatikan bahwa ketika Anda menetapkan prioritas untuk jenis instans untuk kapasitas Spot Anda, prioritas yang sama juga diterapkan ke On-Demand Instans Anda jika strategi On-Demand alokasi disetel ke prioritas. Untuk Spot Fleet, penggunaan prioritas hanya didukung jika armada Anda menggunakan template peluncuran.

Jika Anda menggunakan AWS CLI, nama parameternya adalah `capacity-optimized` dan `capacity-optimized-prioritized` untuk Armada EC2 dan `capacityOptimized` dan `capacityOptimizedPrioritized` untuk Armada Spot.

Diversifikasi

Instans Spot didistribusikan di semua kolam kapasitas Spot. Jika Anda menggunakan AWS CLI, nama parameternya adalah `diversified` untuk Armada EC2 dan Armada Spot.

Harga terendah (tidak disarankan)

> **Warning**
>
> Kami tidak merekomendasikan strategi alokasi harga terendah karena strategi ini memiliki risiko gangguan tertinggi untuk Instans Spot Anda.

Instans Spot berasal dari kolam dengan harga terendah yang memiliki kapasitas tersedia. Saat menggunakan AWS CLI, ini adalah strategi default. Namun, sebaiknya Anda mengganti default dengan menentukan strategi alokasi kapasitas harga yang dioptimalkan.

Dengan strategi harga terendah, jika kumpulan dengan harga terendah tidak memiliki kapasitas yang tersedia, Instans Spot berasal dari kumpulan dengan harga terendah berikutnya yang memiliki kapasitas tersedia. Jika kumpulan kehabisan kapasitas sebelum memenuhi kapasitas yang Anda inginkan, armada akan terus memenuhi permintaan Anda dengan mengambil dari kumpulan harga terendah berikutnya. Untuk memastikan bahwa kapasitas yang Anda inginkan terpenuhi, Anda mungkin menerima Instans Spot dari beberapa kolam.

Karena strategi ini hanya mempertimbangkan harga instans dan bukan ketersediaan kapasitas, hal ini dapat menyebabkan tingkat interupsi yang tinggi.

Strategi alokasi harga terendah hanya tersedia saat menggunakan AWS CLI. Nama parameter adalah `lowest-price` untuk Armada EC2 dan `lowestPrice` untuk Armada Spot.

Jumlah kolam yang akan digunakan

Anda dapat menentukan jumlah kolam Spot untuk mengalokasikan kapasitas Spot target. Berlaku hanya jika strategi alokasi diatur ke harga terendah. Armada memilih kumpulan Spot dengan harga terendah dan mengalokasikan kapasitas Spot target Anda secara merata di seluruh jumlah kumpulan Spot yang Anda tentukan.

Perhatikan bahwa armada mencoba menarik Instans Spot dari jumlah kumpulan yang Anda tentukan berdasarkan upaya terbaik. Jika kumpulan kehabisan kapasitas Spot sebelum memenuhi kapasitas target Anda, armada akan terus memenuhi permintaan Anda dengan menarik dari kumpulan harga terendah berikutnya. Untuk memastikan bahwa kapasitas target terpenuhi, Anda mungkin menerima Instans Spot dari kolam yang jumlahnya lebih dari jumlah kolam yang Anda tentukan. Demikian pula, jika sebagian besar kolam tidak memiliki kapasitas Spot, Anda mungkin menerima kapasitas target penuh dari jumlah yang lebih rendah dari kolam yang Anda tentukan.

Parameter ini hanya tersedia saat menentukan strategi alokasi harga terendah dan hanya saat menggunakan AWS CLI. Nama parameter adalah `InstancePoolsToUseCount` untuk Armada EC2 dan Armada Spot.

## Strategi alokasi untuk On-Demand Instans

Konfigurasi peluncuran menentukan semua kumpulan kapasitas yang mungkin (jenis instans dan Zona Ketersediaan) dari mana Armada EC2 atau Spot Fleet dapat meluncurkan On-Demand Instans. Namun, saat meluncurkan instance, armada menggunakan strategi alokasi yang Anda tentukan untuk memilih kumpulan tertentu dari semua kumpulan yang mungkin.

Anda dapat menentukan salah satu strategi alokasi berikut untuk On-Demand Instans:

Harga terendah

On-Demand Instans berasal dari kumpulan harga terendah yang memiliki kapasitas yang tersedia. Ini adalah strategi default.

Jika kumpulan dengan harga terendah tidak memiliki kapasitas yang tersedia, On-Demand Instans berasal dari kumpulan dengan harga terendah berikutnya yang memiliki kapasitas tersedia.

Jika kumpulan kehabisan kapasitas sebelum memenuhi kapasitas yang Anda inginkan, armada akan terus memenuhi permintaan Anda dengan mengambil dari kumpulan harga terendah berikutnya. Untuk memastikan kapasitas yang Anda inginkan terpenuhi, Anda mungkin menerima On-Demand Instans dari beberapa kumpulan.

Diprioritaskan

Armada menggunakan prioritas yang Anda tetapkan untuk setiap penggantian template peluncuran, meluncurkan jenis instans dalam urutan prioritas tertinggi terlebih dahulu. Strategi ini tidak dapat digunakan dengan pemilihan tipe instance berbasis atribut. Untuk contoh cara menggunakan strategi alokasi ini, lihatPrioritaskan jenis instans untuk On-Demand kapasitas.

## Pilih strategi alokasi Spot yang sesuai

Anda dapat mengoptimalkan armada untuk kasus penggunaan dengan memilih strategi alokasi Spot yang sesuai.

### Menyeimbangkan harga terendah dan ketersediaan kapasitas

Untuk menyeimbangkan pertukaran antara kumpulan kapasitas Spot dengan harga terendah dan kumpulan kapasitas Spot dengan ketersediaan kapasitas tertinggi, sebaiknya gunakan strategi alokasi kapasitas harga yang dioptimalkan. Strategi ini membuat keputusan terkait kolam yang akan meminta Instans Spot dari berdasarkan harga kolam dan ketersediaan kapasitas Instans Spot di kolam tersebut. Ini berarti Instans Spot diminta dari kumpulan yang kemungkinan besar memiliki kemungkinan terganggu terendah dalam waktu dekat, sambil tetap mempertimbangkan harga.

Jika armada Anda menjalankan beban kerja yang tangguh dan tanpa status, termasuk aplikasi kontainer, layanan mikro, aplikasi web, pekerjaan data dan analitik, serta pemrosesan batch, maka gunakan strategi alokasi kapasitas harga yang dioptimalkan untuk penghematan biaya dan ketersediaan kapasitas yang optimal.

Jika armada Anda menjalankan beban kerja yang mungkin memiliki biaya interupsi yang lebih tinggi terkait dengan memulai kembali tugas, Anda harus menerapkan operasi titik pemeriksaan agar aplikasi dapat memulai kembali dari titik tersebut jika terinterupsi. Dengan menggunakan checkpointing, Anda membuat strategi alokasi kapasitas harga yang dioptimalkan cocok untuk beban kerja ini karena mengalokasikan kapasitas dari kumpulan dengan harga terendah yang juga menawarkan tingkat gangguan Instans Spot rendah.

Misalnya konfigurasi JSON yang menggunakan strategi alokasi kapasitas harga yang dioptimalkan, lihat berikut ini:

- Armada EC2 - Contoh 10: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas harga
- Armada Spot — Contoh 11: Luncurkan Instans Spot di harga CapacityOptimized armada

### Ketika beban kerja memiliki biaya interupsi yang tinggi

Secara opsional, Anda dapat menggunakan strategi kapasitas yang dioptimalkan jika menjalankan beban kerja yang menggunakan jenis instans dengan harga yang sama, atau jika biaya interupsi sangat signifikan sehingga penghematan biaya tidak memadai dibandingkan dengan peningkatan interupsi yang marjinal. Strategi ini mengalokasikan kapasitas dari kolam kapasitas Spot yang paling banyak tersedia yang menawarkan kemungkinan lebih sedikit interupsi, yang dapat menurunkan biaya keseluruhan beban kerja Anda.

Ketika kemungkinan interupsi harus diminimalkan tetapi preferensi untuk jenis instans tertentu penting, Anda dapat menyatakan prioritas kumpulan Anda dengan menggunakan strategi alokasi

prioritas yang dioptimalkan kapasitas dan kemudian menetapkan urutan jenis instans yang akan digunakan dari prioritas tertinggi ke terendah.

Perhatikan bahwa ketika Anda menetapkan prioritas untuk prioritas yang dioptimalkan kapasitas, prioritas yang sama juga diterapkan ke On-Demand Instans Anda jika strategi On-Demand alokasi disetel ke prioritas. Perhatikan juga bahwa, untuk Armada Spot, penggunaan prioritas hanya didukung jika armada Anda menggunakan template peluncuran.

Misalnya konfigurasi JSON yang menggunakan strategi alokasi kapasitas yang dioptimalkan, lihat berikut ini:

- Armada EC2 - Contoh 8: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas
- Armada Spot — Contoh 9: Meluncurkan Instans Spot dalam armada yang dioptimalkan kapasitas

Misalnya konfigurasi JSON yang menggunakan strategi alokasi prioritas yang dioptimalkan kapasitas, lihat berikut ini:

- Armada EC2 - Contoh 9: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas
- Armada Spot — Contoh 10: Meluncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas

Jika beban kerja Anda memiliki fleksibilitas waktu dan ketersediaan kapasitas tidak menjadi faktor

Jika armada Anda kecil atau berjalan untuk waktu yang singkat, Anda dapat menggunakan kapasitas harga yang dioptimalkan untuk memaksimalkan penghematan biaya sambil tetap mempertimbangkan ketersediaan kapasitas.

Jika armada Anda besar atau berjalan untuk waktu yang lama

Jika armada Anda besar atau berjalan untuk waktu yang lama, Anda dapat meningkatkan ketersediaan armada Anda dengan mendistribusikan Instans Spot ke beberapa kumpulan menggunakan strategi yang beragam. Misalnya, jika armada Anda menentukan 10 kumpulan dan kapasitas target 100 instans, armada akan meluncurkan 10 Instans Spot di setiap kumpulan. Jika harga Spot untuk satu kolam melebihi harga maksimum Anda untuk kolam ini, hanya 10% armada yang terpengaruh. Penggunaan strategi ini juga membuat armada Anda kurang sensitif terhadap kenaikan harga Spot di satu kolam dari waktu ke waktu. Dengan strategi yang beragam, armada

tidak meluncurkan Instans Spot ke kumpulan mana pun dengan harga Spot yang sama atau lebih tinggi dari On-Demand harga.

## Mempertahankan kapasitas target untuk Instans Spot

Setelah Instans Spot dihentikan karena perubahan harga Spot atau kapasitas yang tersedia dari kumpulan kapasitas Spot, armada jenis `maintain` meluncurkan Instans Spot pengganti. Strategi alokasi menentukan kolam tempat instans pengganti diluncurkan, sebagai berikut:

- Jika strategi alokasi dioptimalkan kapasitas harga, armada akan meluncurkan instans pengganti di kumpulan yang memiliki ketersediaan kapasitas Instans Spot terbanyak sekaligus mempertimbangkan harga dan mengidentifikasi kumpulan dengan harga terendah dengan ketersediaan kapasitas tinggi.
- Jika strategi alokasi dioptimalkan kapasitas, armada akan meluncurkan instans pengganti di kumpulan yang memiliki ketersediaan kapasitas Instans Spot paling banyak.
- Jika strategi alokasi ter diversifikasi, armada akan mendistribusikan Instans Spot pengganti ke seluruh kumpulan yang tersisa.

## Prioritaskan jenis instans untuk On-Demand kapasitas

Ketika Armada EC2 atau Spot Fleet mencoba memenuhi On-Demand kapasitas Anda, secara default akan meluncurkan jenis instans dengan harga terendah terlebih dahulu. Jika strategi On-Demand alokasi diatur ke prioritas, armada menggunakan prioritas untuk menentukan jenis instans mana yang akan digunakan pertama kali saat memenuhi On-Demand kapasitas. Prioritas ditetapkan ke penimpaan templat peluncuran, dan prioritas tertinggi diluncurkan terlebih dahulu.

Contoh: Memprioritaskan tipe instans

Dalam contoh ini, Anda mengonfigurasi tiga penimpaan templat peluncuran, masing-masing dengan tipe instans yang berbeda.

On-Demand Harga untuk jenis instance berkisar dalam harga. Berikut ini adalah tipe instans yang digunakan dalam contoh ini, yang tercantum dalam urutan harga, dimulai dengan tipe instans yang paling murah:

- `m4.large` – termurah
- `m5.large`
- `m5a.large`

Jika Anda tidak menggunakan prioritas untuk menentukan urutan, armada memenuhi On-Demand kapasitas dengan memulai dengan jenis instans yang paling murah.

Namun, katakanlah Anda memiliki Instans Terpesan `m5.large` yang tidak terpakai yang ingin Anda gunakan terlebih dahulu. Anda dapat mengatur prioritas penimpaan templat peluncuran sehingga tipe instans digunakan dalam urutan prioritas, sebagai berikut:

- `m5.large` – prioritas 1
- `m4.large` – prioritas 2
- `m5a.large` – prioritas 3

## Gunakan Penyeimbangan Kembali Kapasitas di Armada EC2 dan Spot Fleet untuk menggantikan Instans Spot yang berisiko

Dengan Capacity Rebalancing, Armada EC2 atau Spot Fleet Anda dapat mempertahankan kapasitas Spot yang diinginkan dengan secara proaktif mengganti Instans Spot yang berisiko terganggu. Ketika Instans Spot berada pada risiko gangguan yang tinggi, Amazon EC2 mengirimkan rekomendasi rebalance. Jika Penyeimbangan Kembali Kapasitas diaktifkan, rekomendasi rebalance akan memicu peluncuran Instans Spot baru sebelum instans berisiko terganggu.

Penyeimbangan Kembali Kapasitas membantu Anda menjaga ketersediaan beban kerja dengan secara proaktif menambah armada Anda dengan Instans Spot baru sebelum instans yang berjalan terganggu oleh Amazon EC2.

Untuk mengonfigurasi EC2 Fleet agar menggunakan Capacity Rebalancing untuk meluncurkan Instans Spot pengganti

Gunakan perintah create-fleet dan parameter yang relevan dalam struktur. `MaintenanceStrategies` Untuk contoh konfigurasi JSON, lihatContoh 7: Konfigurasikan Penyeimbangan Kembali Kapasitas untuk meluncurkan Instans Spot pengganti.

Untuk mengonfigurasi Spot Fleet agar menggunakan Penyeimbangan Kembali Kapasitas untuk meluncurkan Instans Spot pengganti

Anda dapat menggunakan konsol Amazon EC2 atau AWS CLI untuk mengonfigurasi Penyeimbangan Kembali Kapasitas.

(Konsol) Saat membuat Spot Fleet, pilih kotak centang Penye imbang kembali kapasitas. Untuk informasi selengkapnya, lihat langkah 6.d. di Membuat permintaan Spot Fleet menggunakan parameter yang ditentukan.

(AWS CLI) Gunakan perintah request-spot-fleet dan parameter yang relevan dalam struktur. `SpotMaintenanceStrategies` Untuk contoh konfigurasi JSON, lihatContoh 8: Mengonfigurasikan Penyeimbangan Ulang Kapasitas untuk meluncurkan Instans Spot pengganti.

Topik

- Batasan
- Opsi konfigurasi
- Pertimbangan-pertimbangan

## Batasan

- Penyeimbangan Ulang Kapasitas hanya tersedia untuk armada tipe `maintain`.
- Saat armada berjalan, Anda tidak dapat mengubah pengaturan Penyeimbangan Ulang Kapasitas. Untuk mengubah pengaturan Penyeimbangan Ulang Kapasitas, Anda harus menghapus armada dan membuat armada baru.

## Opsi konfigurasi

Ar `ReplacementStrategy` mada EC2 dan Armada Spot mendukung dua nilai berikut:

`launch-before-terminate`

Amazon EC2 mengakhiri Instans Spot yang menerima notifikasi penyeimbangan ulang setelah Instans Spot pengganti baru diluncurkan. Jika Anda menentukan `launch-before-terminate`, Anda juga harus menentukan nilai untuk `termination-delay`. Setelah instans pengganti baru diluncurkan, Amazon EC2 menunggu durasi `termination-delay`, lalu mengakhiri instans lama. Untuk `termination-delay`, minimum adalah 120 detik (2 menit), dan maksimum adalah 7200 detik (2 jam).

Sebaiknya Anda menggunakan `launch-before-terminate` hanya jika Anda dapat memprediksi lamanya prosedur pematian instans Anda akan selesai. Hal ini akan memastikan bahwa instans lama diakhiri hanya setelah prosedur pematian selesai. Perhatikan bahwa Amazon

EC2 dapat menginterupsi instans lama dengan peringatan dua menit sebelum `termination-delay`.

Kami sangat menyarankan agar tidak menggunakan strategi alokasi `lowest-price` (Armada EC2) atau `lowestPrice` (Spot Fleet) dalam kombinasi dengan `launch-before-terminate` menghindari Instans Spot pengganti yang juga berisiko tinggi mengalami gangguan.

`launch`

Amazon EC2 meluncurkan Instans Spot pengganti saat notifikasi penyeimbangan ulang dipancarkan untuk Instans Spot yang sudah ada. Amazon EC2 tidak mengakhiri instans yang menerima notifikasi penyeimbangan ulang. Anda dapat mengakhiri instans lama, atau membiarkannya berjalan. Anda dikenai biaya untuk semua instans saat semuanya berjalan.

## Pertimbangan-pertimbangan

Jika Anda mengonfigurasi Armada EC2 atau Armada Spot untuk Penyeimbangan Kembali Kapasitas, pertimbangkan hal berikut:

Berikan sebanyak mungkin kolam kapasitas Spot dalam permintaan

Konfigurasikan armada Anda untuk menggunakan beberapa jenis instans dan Zona Ketersediaan. Hal ini akan memberikan fleksibilitas untuk meluncurkan Instans Spot di berbagai kolam kapasitas Spot. Untuk informasi selengkapnya, lihat Bersikaplah fleksibel terkait tipe instans dan Zona Ketersediaan.

Hindari peningkatan risiko gangguan penggantian Instans Spot

Untuk menghindari peningkatan risiko gangguan, kami merekomendasikan strategi `capacity-optimized-prioritized` alokasi `capacity-optimized` or. Strategi ini memastikan bahwa Instans Spot diluncurkan di kolam kapasitas Spot yang paling optimal, dan karena itu kemungkinan tidak akan terinterupsi dalam waktu dekat. Untuk informasi selengkapnya, lihat Menggunakan strategi alokasi harga dan kapasitas yang dioptimalkan.

Jika Anda menggunakan strategi al `lowest-price` okasi, Instans Spot pengganti Anda mungkin berisiko lebih tinggi mengalami gangguan. Hal ini disebabkan karena Amazon EC2 akan selalu meluncurkan instans di kolam dengan harga terendah yang memiliki kapasitas yang tersedia pada saat itu, meskipun Instans Spot pengganti Anda kemungkinan akan terinterupsi sesaat setelah diluncurkan.

Amazon EC2 hanya akan meluncurkan instans baru jika ketersediaannya sama atau lebih baik

Salah satu tujuan dari Penyeimbangan Ulang kapasitas adalah untuk meningkatkan ketersediaan Instans Spot. Jika Instans Spot yang ada menerima rekomendasi penyeimbangan ulang, Amazon EC2 hanya akan meluncurkan instans baru jika instans baru tersebut memberikan ketersediaan yang sama atau lebih baik daripada instans yang sudah ada. Jika risiko gangguan instans baru akan lebih buruk daripada instans yang sudah ada, Amazon EC2 tidak akan meluncurkan instans baru. Namun, Amazon EC2 akan terus menilai kolam kapasitas Spot, dan akan meluncurkan instans baru jika ketersediaan membaik.

Ada kemungkinan instans Anda yang ada akan terinterupsi tanpa Amazon EC2 yang secara proaktif meluncurkan instans baru. Jika hal ini terjadi, Amazon EC2 akan berupaya meluncurkan instans baru terlepas dari apakah instans baru tersebut memiliki risiko gangguan yang tinggi.

Penyeimbangan Ulang Kapasitas tidak meningkatkan tingkat interupsi Instans Spot Anda

Saat Anda mengaktifkan Penyeimbangan Ulang Kapasitas, hal tersebut tidak meningkatkan tingkat interupsi Instans Spot Anda (jumlah Instans Spot yang diklaim kembali saat Amazon EC2 membutuhkan kapasitas kembali). Namun, jika Penyeimbangan Ulang Kapasitas mendeteksi instans yang berada pada berisiko terinterupsi, Amazon EC2 akan segera berupaya meluncurkan instans baru. Hasilnya adalah lebih banyak instans yang mungkin diganti dibandingkan jika Anda menunggu Amazon EC2 meluncurkan instans baru setelah instans yang berisiko terinterupsi.

Meskipun Anda dapat mengganti lebih banyak instans dengan Penyeimbangan Ulang Kapasitas diaktifkan, Anda akan mendapatkan keuntungan dengan bersikap proaktif daripada reaktif dengan memiliki lebih banyak waktu untuk mengambil tindakan sebelum instans Anda terinterupsi. Dengan pemberitahuan interupsi Instans Spot, Anda biasanya hanya memiliki waktu hingga dua menit untuk mematikan instans Anda dengan baik. Dengan Penyeimbangan Ulang Kapasitas meluncurkan instans baru terlebih dahulu, Anda memberikan kesempatan yang lebih baik untuk menyelesaikan proses yang sudah ada pada instans berisiko, Anda dapat memulai prosedur pematian instans, dan mencegah pekerjaan baru dijadwalkan pada instans berisiko Anda. Anda juga bisa mulai menyiapkan instans yang baru diluncurkan untuk mengambil alih aplikasi. Dengan penggantian proaktif dari Penyeimbangan Ulang Kapasitas, Anda akan mendapatkan keuntungan dari kesinambungan yang baik.

Sebagai contoh teoretis untuk menunjukkan risiko dan manfaat menggunakan Penyeimbangan Ulang Kapasitas, pertimbangkan skenario berikut:

- 14:00 – Rekomendasi penyeimbangan ulang diterima untuk instans-A, dan Amazon EC2 segera mulai berupaya meluncurkan instans-B pengganti, sehingga memberi Anda waktu untuk memulai prosedur pematian.*
- 14:30 – Rekomendasi penyeimbangan ulang diterima untuk instans-B, diganti dengan instans-C, sehingga memberi Anda waktu untuk memulai prosedur pematian.*
- 14:32 — Jika Penyeimbangan Kembali Kapasitas tidak diaktifkan, dan jika pemberitahuan gangguan Instans Spot diterima pada pukul 14:32 untuk Instance-A, Anda hanya memiliki waktu hingga dua menit untuk mengambil tindakan, tetapi Instance-A akan berjalan hingga saat ini.

* Jika `launch-before-terminate` ditentukan, Amazon EC2 akan mengakhiri instans yang berada dalam risiko setelah instans pengganti online.

Amazon EC2 dapat meluncurkan Instans Spot pengganti yang baru hingga kapasitas yang terpenuhi adalah dua kali lipat dari kapasitas target

Saat armada dikonfigurasi untuk Penyeimbangan Kembali Kapasitas, armada akan mencoba meluncurkan Instans Spot pengganti baru untuk setiap Instans Spot yang menerima rekomendasi rebalance. Setelah Instans Spot menerima rekomendasi penyeimbangan ulang, Instans Spot tersebut tidak lagi dianggap sebagai bagian dari kapasitas yang terpenuhi. Bergantung pada strategi penggantian, Amazon EC2 akan mengakhiri instans setelah penundaan pengakhiran yang telah dikonfigurasikan sebelumnya, atau membiarkannya tetap berjalan. Hal ini memberikan kesempatan kepada Anda untuk melakukan tindakan penyeimbangan ulang pada instans.

Jika armada Anda mencapai dua kali lipat dari kapasitas target, armada akan berhenti meluncurkan instans pengganti yang baru meskipun instans pengganti itu sendiri menerima rekomendasi penyeimbangan ulang.

Misalnya, Anda membuat armada dengan kapasitas target 100 Instans Spot. Semua Instans Spot menerima rekomendasi penyeimbangan ulang, yang menyebabkan Amazon EC2 meluncurkan 100 Instans Spot pengganti. Hal ini meningkatkan jumlah Instans Spot yang terpenuhi menjadi 200, atau dua kali lipat dari kapasitas yang ditargetkan. Beberapa instans pengganti menerima rekomendasi rebalance, tetapi tidak ada lagi instans pengganti yang diluncurkan karena armada tidak dapat melebihi dua kali lipat kapasitas targetnya.

Perhatikan bahwa Anda dikenai biaya untuk semua instans saat berjalan.

Sebaiknya Anda mengonfigurasi armada untuk menghentikan Instans Spot yang menerima rekomendasi rebalance

Jika Anda mengonfigurasi armada untuk Penyeimbangan Kembali Kapasitas, sebaiknya pilih `launch-before-terminate` dengan penundaan penghentian yang sesuai hanya jika Anda dapat memprediksi berapa lama prosedur penutupan instans akan diselesaikan. Hal ini akan memastikan bahwa instans lama diakhiri hanya setelah prosedur pematian selesai.

Jika memilih untuk mengakhiri instans yang direkomendasikan untuk penyeimbangan ulang, kami menyarankan Anda untuk memantau sinyal rekomendasi penyeimbangan ulang yang diterima oleh Instans Spot di armada. Dengan memantau sinyal, Anda dapat dengan cepat melakukan tindakan penyeimbangan ulang pada instans yang terpengaruh sebelum Amazon EC2 menginterupsinya, lalu Anda dapat mengakhirinya secara manual. Jika Anda tidak mengakhiri instans tersebut, Anda akan terus membayarnya saat instans tersebut berjalan. Amazon EC2 tidak secara otomatis mengakhiri instans yang menerima notifikasi penyeimbangan ulang.

Anda dapat mengatur notifikasi menggunakan Amazon EventBridge atau metadata instans. Untuk informasi selengkapnya, lihat Pantau sinyal rekomendasi penyeimbangan kembali.

Armada tidak menghitung instans yang menerima rekomendasi rebalance saat menghitung kapasitas yang terpenuhi selama skala masuk atau keluar

Jika armada Anda dikonfigurasi untuk Penyeimbangan Kembali Kapasitas, dan Anda mengubah kapasitas target untuk menskalakan masuk atau menskalakan, armada tidak menghitung instans yang ditandai untuk rebalance sebagai bagian dari kapasitas yang terpenuhi, sebagai berikut:

- Menskalakan ke dalam – Jika Anda menurunkan kapasitas target yang Anda inginkan, Amazon EC2 akan mengakhiri instans yang tidak ditandai untuk penyeimbangan ulang hingga kapasitas yang diinginkan tercapai. Instans yang ditandai untuk penyeimbangan ulang tidak diperhitungkan pada kapasitas yang terpenuhi.

  Misalnya, Anda membuat armada dengan kapasitas target 100 Instans Spot. 10 instans menerima rekomendasi rebalance, sehingga Amazon EC2 meluncurkan 10 instans pengganti baru, menghasilkan kapasitas terpenuhi 110 instans. Anda kemudian mengurangi kapasitas target menjadi 50 (menskalakan ke dalam), tetapi kapasitas yang terpenuhi sebenarnya adalah 60 instans karena 10 instans yang ditandai untuk penyeimbangan ulang tidak diakhiri oleh Amazon EC2. Anda harus mengakhiri instans ini secara manual, atau Anda dapat membiarkannya tetap berjalan.

- Menskalakan ke luar – Jika Anda meningkatkan kapasitas target yang diinginkan, Amazon EC2 akan meluncurkan instans baru hingga kapasitas yang diinginkan tercapai. Instans yang ditandai untuk penyeimbangan ulang tidak diperhitungkan pada kapasitas yang terpenuhi.

  Misalnya, Anda membuat armada dengan kapasitas target 100 Instans Spot. 10 instans menerima rekomendasi rebalance, sehingga armada meluncurkan 10 instans pengganti baru, sehingga kapasitas terpenuhi 110 instans. Anda kemudian meningkatkan kapasitas target menjadi 200 (menskalakan ke luar), tetapi kapasitas yang terpenuhi sebenarnya adalah 210 instans karena 10 instans yang ditandai untuk penyeimbangan ulang tidak diperhitungkan oleh armada sebagai bagian dari kapasitas target. Anda harus mengakhiri instans ini secara manual, atau Anda dapat membiarkannya tetap berjalan.

## Gunakan Reservasi Kapasitas untuk cadangan On-Demand kapasitas di Armada EC2

Dengan Reservasi On-Demand Kapasitas, Anda dapat memesan kapasitas komputasi untuk On-Demand Instans Anda di Zona Ketersediaan tertentu untuk durasi apa pun. Anda dapat mengonfigurasi Armada EC2 untuk menggunakan Reservasi Kapasitas terlebih dahulu saat meluncurkan On-Demand Instans.

On-Demand Reservasi Kapasitas hanya tersedia untuk Armada EC2 dengan jenis permintaan disetel ke`instant`.

Reservasi Kapasitas dikonfigurasi sebagai `open` atau `targeted`. EC2 Fleet dapat meluncurkan On-Demand Instans ke salah satu `open` atau Reserv `targeted` asi Kapasitas, sebagai berikut:

- Jika Reservasi Kapasitas adalah`open`, On-Demand Instans yang memiliki atribut yang cocok secara otomatis berjalan dalam kapasitas yang dicadangkan.
- Jika Reservasi Kapasitas adalah`targeted`, On-Demand Instans harus menargetkannya secara khusus agar berjalan dalam kapasitas yang dicadangkan. Hal ini berguna untuk menggunakan Reservasi Kapasitas tertentu atau untuk mengontrol kapan harus menggunakan Reservasi Kapasitas tertentu.

Jika Anda menggunakan Reservasi `targeted` Kapasitas di Armada EC2 Anda, harus ada Reservasi Kapasitas yang cukup untuk memenuhi On-Demand kapasitas target, jika tidak peluncuran gagal. Untuk menghindari kegagalan peluncuran, lebih baik tambahkan Reservasi Kapasitas `targeted` ke grup sumber daya, lalu targetkan grup sumber daya tersebut. Grup sumber daya tidak perlu

memiliki Reservasi Kapasitas yang cukup; jika kehabisan Reservasi Kapasitas sebelum On-Demand kapasitas target terpenuhi, armada dapat meluncurkan kapasitas target yang tersisa ke On-Demand kapasitas reguler.

Untuk menggunakan Reservasi Kapasitas dengan Armada EC2

1. Konfigurasikan armada sebagai tipe `instant`. Anda tidak dapat menggunakan Reservasi Kapasitas untuk armada tipe lain.
2. Konfigurasikan strategi penggunaan untuk Reservasi Kapasitas sebagai `use-capacity-reservations-first`.
3. Pada templat peluncuran, untuk Reservasi kapasitas, pilih Buka atau Target berdasarkan grup. Jika Anda memilih Target berdasarkan grup, tentukan ID grup sumber daya Reservasi Kapasitas.

Ketika armada mencoba memenuhi On-Demand kapasitas, jika menemukan bahwa beberapa kumpulan instans memiliki Reservasi Kapasitas pencocokan yang tidak digunakan, armada akan menentukan kumpulan tempat peluncuran On-Demand Instans berdasarkan strategi On-Demand alokasi (`lowest-price`atau`prioritized`).

Sumber daya terkait

- Untuk contoh CLI tentang cara mengonfigurasi armada untuk menggunakan Reservasi Kapasitas untuk memenuhi On-Demand kapasitas, lihatContoh konfigurasi CLI untuk Armada EC2, khususnya Contoh 5 hingga 7.
- Untuk tutorial yang memandu Anda melalui langkah-langkah untuk membuat Reservasi Kapasitas, menggunakannya di armada Anda, dan melihat berapa banyak Reservasi Kapasitas yang tersisa, lihat Tutorial: Mengkonfigurasi Armada EC2 untuk meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan
- Anda juga dapat mengonfigurasi Armada EC2 untuk meluncurkan instans di beberapa jenis Reservasi Kapasitas — Reservasi On-Demand Kapasitas, Blok Kapasitas untuk ML, dan Reservasi Kapasitas yang dapat diinterupsi — menggunakan Grup Sumber Daya Reservasi Kapasitas tunggal, dan secara opsional kembali ke On-Demand kapasitas. Untuk melakukan ini, gunakan `ReservedCapacityOptions` dengan `DefaultTargetCapacityType` set ke`reserved-capacity`. Untuk tutorial, lihat Tutorial: Mengkonfigurasi Armada EC2 Anda untuk meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas.

- Untuk informasi tentang mengonfigurasi Reservasi Kapasitas, lihat Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand dan FAQ Reservasi On-Demand Kapasitas.

# Bekerja dengan EC2 Fleet

Untuk mulai menggunakan Armada EC2, buat permintaan yang mencakup total kapasitas target, kapasitas, On-Demand kapasitas Spot, dan template peluncuran yang menentukan konfigurasi untuk instans dalam armada. Anda dapat secara opsional menentukan parameter tambahan, atau membiarkan armada menggunakan nilai default. Anda juga dapat menandai permintaan armada, serta instance serta volumenya, saat membuat armada.

Armada akan meluncurkan On-Demand Instans jika ada kapasitas yang tersedia, dan meluncurkan Instans Spot saat harga maksimum melebihi harga Spot dan kapasitas yang tersedia.

Setelah armada Anda diluncurkan, Anda dapat menjelaskan permintaan armada, instance dalam armada, dan peristiwa armada apa pun. Anda juga dapat menetapkan tag tambahan sesuai kebutuhan.

Jika Anda perlu mengubah parameter armada apa pun, seperti kapasitas target total, Anda dapat memodifikasi armada, asalkan dikonfigurasi untuk mempertahankan kapasitas. Anda tidak dapat mengubah kapasitas permintaan satu kali setelah dikirimkan.

Permintaan armada tetap aktif hingga kedaluwarsa atau Anda menghapusnya. Saat menghapus permintaan armada, Anda dapat menghentikan instans atau membiarkannya berjalan. Jika Anda memilih untuk membiarkannya berjalan, On-Demand Instans akan berjalan sampai Anda menghentikannya, dan Instans Spot berjalan hingga terganggu atau Anda menghentikannya.

Topik


- Status permintaan Armada EC2
- Prasyarat Armada EC2
- Membuat Armada EC2
- Menandai permintaan Armada EC2 yang baru atau yang sudah ada serta instans serta volume yang diluncurkan
- Jelaskan Armada EC2, instansnya, dan kejadiannya
- Memodifikasi Armada EC2
- Menghapus permintaan Armada EC2 dan instans di armada

# Status permintaan Armada EC2

Permintaan Armada EC2 dapat berupa salah satu dari berbagai status, dengan setiap status menunjukkan tahap yang berbeda dari siklus hidup permintaan dan pengelolaan instansnya.

Permintaan Armada EC2 dapat berada dalam salah satu status berikut:

`submitted`

Permintaan Armada EC2 sedang dievaluasi dan Amazon EC2 sedang bersiap untuk meluncurkan jumlah target instans. Jika permintaan melebihi batas armada Anda, permintaan akan segera dihapus.

`active`

Armada EC2 telah divalidasi dan Amazon EC2 berupaya untuk mempertahankan jumlah target dari instans yang sedang berjalan. Permintaan tetap berada dalam status ini sampai dimodifikasi atau dihapus.

`modifying`

Permintaan Armada EC2 sedang dimodifikasi. Permintaan tetap berada dalam status ini sampai modifikasi diproses sepenuhnya atau permintaan dihapus. Hanya tipe armada `maintain` yang dapat dimodifikasi. Status ini tidak berlaku untuk tipe permintaan lain.

`deleted_running`

Permintaan EC2 Fleet dihapus dan tidak meluncurkan Instans Spot tambahan. Instans yang ada terus berjalan sampai diinterupsi atau diakhiri secara manual. Permintaan tetap berada dalam status ini sampai semua instans diinterupsi atau diakhiri. Hanya tipe Armada EC2 `maintain` atau `request` yang dapat menjalankan instans setelah permintaan Armada EC2 dihapus. Armada `instant` yang dihapus dengan instans yang sedang berjalan tidak didukung. Status ini tidak berlaku untuk armada `instant`.

`deleted_terminating`

Permintaan Armada EC2 dihapus dan instansnya diakhiri. Permintaan tetap dalam keadaan ini sampai semua contoh dihentikan.

`deleted`

Permintaan Armada EC2 dihapus dan tidak ada instans yang berjalan. Permintaan tersebut dihapus dua hari setelah instansnya diakhiri.

# Prasyarat Armada EC2

Untuk membuat Armada EC2, prasyarat berikut harus ada:

- Luncurkan templat
- Service-linked peran untuk Armada EC2
- Memberikan akses ke kunci yang dikelola pelanggan untuk digunakan dengan AMI terenkripsi dan snapshot EBS
- Izin untuk pengguna Armada EC2

## Luncurkan templat

Template peluncuran menentukan informasi konfigurasi tentang instans yang akan diluncurkan, seperti jenis instans dan Zona Ketersediaan. Untuk informasi selengkapnya tentang template peluncuran, lihatSimpan parameter peluncuran instans di templat peluncuran Amazon EC2.

## Service-linked peran untuk Armada EC2

Peran `AWSServiceRoleForEC2Fleet` memberikan izin kepada Armada EC2 untuk meminta, meluncurkan, mengakhiri, dan menandai instans atas nama Anda. Amazon EC2 menggunakan peran tertaut layanan ini untuk menyelesaikan tindakan berikut:

- `ec2:RunInstances` – Meluncurkan instans.
- `ec2:RequestSpotInstances` – Meminta Instans Spot.
- `ec2:TerminateInstances` – Mengakhiri instans.
- `ec2:DescribeImages`— Jelaskan Amazon Machine Images (AMI) untuk instance.
- `ec2:DescribeInstanceStatus`— Jelaskan status instance.
- `ec2:DescribeSubnets`— Jelaskan subnet untuk instance.
- `ec2:CreateTags` – Menambahkan tanda ke Armada, instans, dan volume EC2.

Pastikan peran ini ada sebelum Anda menggunakan AWS CLI atau API untuk membuat Armada EC2.

> **Note**
>
> Armada EC2 `instant` tidak membutuhkan peran ini.

Untuk membuat peran, gunakan konsol IAM sebagai berikut.

Untuk membuat AWSServiceRoleForEC2Fleet peran untuk Armada EC2

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Pilih Buat peran.
4. Pada halaman Pilih entitas tepercaya, lakukan hal berikut:
   a. Untuk Ti pe entitas tepercaya, pilih AWS layanan.
   b. Di bawah Kasus penggunaan, untuk Layanan atau kasus penggunaan, pilih EC2 - Armada.

      > **Tip**
      >
      > Pastikan untuk memilih EC2 - Armada. Jika Anda memilih EC2, kasus penggunaan EC2 - Armada tidak muncul dalam daftar kasus penggunaan. Kas us penggunaan EC2 - Fleet akan secara otomatis membuat kebijakan dengan izin IAM yang diperlukan dan akan menyarankan AWSServiceRoleForEC2Fleet sebagai nama peran.

   c. Pilih Berikutnya.
5. Pada halaman Tambahkan izin, pilih Berikutnya.
6. Pada halaman Nama, ulasan, dan buat, pilih Buat peran.

Jika Anda tidak lagi perlu menggunakan EC2 Fleet, kami menyarankan Anda untuk menghapus fileAWSServiceRoleForEC2Fleet wewenang. Setelah peran ini dihapus dari akun Anda, Anda dapat membuat peran tersebut kembali jika Anda membuat armada lain.

Untuk informasi selengkapnya, lihat Service-linked peran di Panduan Pengguna IAM.

## Memberikan akses ke kunci yang dikelola pelanggan untuk digunakan dengan AMI terenkripsi dan snapshot EBS

Jika Anda menentukan AMI terenkripsi atau snapshot Amazon EBS terenkripsi di Armada EC2 Anda dan Anda menggunakan AWS KMS kunci untuk enkripsi, Anda harus memberikan izin AWSServiceRoleForEC2Fleet peran untuk menggunakan kunci yang dikelola pelanggan sehingga

Amazon EC2 dapat meluncurkan instans atas nama Anda. Untuk melakukannya, Anda harus menambahkan pemberian izin ke kunci yang dikelola pelanggan, seperti yang ditunjukkan dalam prosedur berikut.

Ketika memberikan izin, pemberian izin merupakan alternatif dari kebijakan kunci. Untuk informasi selengkapnya, lihat Menggunakan pemberian izin dan Menggunakan kebijakan kunci di AWS KMS dalam Panduan Developer AWS Key Management Service .

Untuk memberikan izin AWSServiceRoleForEC2Fleet peran untuk menggunakan kunci yang dikelola pelanggan

- Gunakan perintah create-grant untuk menambahkan hibah ke kunci yang dikelola pelanggan dan untuk menentukan prinsipal (peran AWSServiceRoleForEC2Fleet terkait layanan) yang diberi izin untuk melakukan operasi yang diizinkan oleh hibah. Kunci yang dikelola pelanggan ditentukan oleh parameter `key-id` dan ARN kunci yang dikelola pelanggan. Kepala sekolah ditentukan oleh`grantee-principal` parameter dan ARN dariAWSServiceRoleForEC2Fleet peran terkait layanan.

```
aws kms create-grant \
    --region us-east-1 \
    --key-id arn:aws:kms:us-
east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/AWSServiceRoleForEC2Fleet \
    --operations "Decrypt" "Encrypt" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
 "ReEncryptTo"
```

## Izin untuk pengguna Armada EC2

Jika pengguna Anda akan membuat atau mengelola Armada EC2, pastikan untuk memberikan izin kepada pengguna.

Untuk membuat kebijakan Armada EC2

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Kebijakan.
3. Pilih Buat kebijakan.

4. Di halaman Buat kebijakan, pilih tab JSON, ganti teks dengan berikut ini, dan pilih Tinjau kebijakan.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
              "iam:ListRoles",
              "iam:PassRole",
              "iam:ListInstanceProfiles"
            ],
            "Resource":"arn:aws:iam::123456789012:role/DevTeam*"
        }
    ]
}
```

   `ec2:*` memberikan izin kepada pengguna untuk memanggil semua tindakan API Amazon EC2. Untuk membatasi pengguna pada tindakan API Amazon EC2 tertentu, tentukan tindakan tersebut.

   Pengguna harus memiliki izin untuk memanggil tindakan `iam:ListRoles` untuk melakukan enumerasi peran IAM yang sudah ada, tindakan `iam:PassRole` untuk menentukan peran Armada EC2, dan tindakan `iam:ListInstanceProfiles` untuk melakukan enumerasi profil instans yang sudah ada.

   (Opsional) Untuk memungkinkan pengguna membuat peran atau profil instans menggunakan konsol IAM, Anda juga harus menambahkan tindakan berikut ke kebijakan:

   - `iam:AddRoleToInstanceProfile`

- `iam:AttachRolePolicy`
- `iam:CreateInstanceProfile`
- `iam:CreateRole`
- `iam:GetRole`
- `iam:ListPolicies`

5. Pada halaman Tinjau kebijakan, masukkan nama dan deskripsi kebijakan, dan pilih Buat kebijakan.
6. Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:
   - Pengguna dan grup di AWS IAM Identity Center:

     Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
   - Pengguna yang dikelola di IAM melalui penyedia identitas:

     Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
   - Pengguna IAM:
     - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
     - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

## Membuat Armada EC2

Untuk membuat Armada EC2, tentukan konfigurasi armada dalam file JSON dan referensi file dengan perintah create- fleet. Dalam file JSON, Anda harus menentukan kapasitas target total untuk armada, kapasitas target terpisah untuk Instans Spot dan On-Demand Instans, dan template peluncuran yang menentukan konfigurasi untuk instans dalam armada, seperti AMI, jenis instans, subnet atau Zona Ketersediaan, dan satu atau beberapa grup keamanan. Secara opsional, Anda dapat menentukan konfigurasi tambahan, seperti parameter untuk mengganti konfigurasi template peluncuran, strategi alokasi untuk memilih Instans Spot dan On-Demand Instans dari kumpulan kapasitas EC2, dan jumlah maksimum yang bersedia Anda bayar untuk armada tersebut. Untuk informasi selengkapnya, lihat Opsi konfigurasi untuk Armada EC2 atau Spot Fleet Anda.

Armada EC2 meluncurkan On-Demand Instans saat kapasitas tersedia, dan meluncurkan Instans Spot saat harga maksimum melebihi harga Spot dan kapasitas yang tersedia.

Jika armada Anda menyertakan Instans Spot dan jenisnya`maintain`, Amazon EC2 akan mencoba mempertahankan kapasitas target armada Anda saat Instans Spot terganggu.

## Batasan Armada EC2

Batasan berikut berlaku untuk Armada EC2:

- Membuat Armada EC2 hanya tersedia melalui Amazon EC2 API,, AWS SDK AWS CLI, dan. CloudFormation
- Permintaan Armada EC2 tidak dapat menjangkau Wil AWS ayah. Anda perlu membuat Armada EC2 terpisah untuk setiap Wilayah.
- Permintaan Armada EC2 tidak dapat menjangkau subnet yang berbeda dari Zona Ketersediaan yang sama.

## Membuat Armada EC2

Untuk meluncurkan armada instans menggunakan EC2 Fleet, Anda hanya perlu menentukan parameter berikut dalam permintaan armada Anda, dan armada akan menggunakan nilai default untuk parameter lainnya:

- `LaunchTemplateId`atau `LaunchTemplateName` — Menentukan template peluncuran yang akan digunakan (yang berisi parameter untuk instance yang akan diluncurkan, seperti jenis instans dan Zona Ketersediaan)
- `TotalTargetCapacity` – Menentukan total kapasitas target untuk armada
- `DefaultTargetCapacityType`— Menentukan apakah opsi pembelian default adalah On-Demand atau Spot

Untuk mengganti parameter yang ditentukan dalam template peluncuran, Anda dapat menentukan satu atau lebih penggantian. Setiap penggantian dapat bervariasi menurut jenis instans, Zona Ketersediaan, subnet, dan harga maksimum, dan dapat menyertakan kapasitas tertimbang yang berbeda. Sebagai alternatif untuk menentukan jenis instans, Anda dapat menentukan atribut yang harus dimiliki instans, dan Amazon EC2 akan mengidentifikasi semua jenis instans dengan atribut tersebut. Untuk informasi selengkapnya, lihat Tentukan atribut untuk pemilihan tipe instans untuk EC2 Fleet atau Spot Fleet.

Untuk tipe Armada EC2`instant`, Anda juga dapat mengganti parameter template peluncuran berikut:

- `IamInstanceProfile`- Profil instance IAM untuk dikaitkan dengan instance. Untuk informasi selengkapnya, lihat IAM role untuk Amazon EC2.
- `KeyName`— Nama pasangan kunci yang akan digunakan untuk instance. Untuk informasi selengkapnya, lihat Pasangan kunci Amazon EC2 dan instans Amazon EC2.
- `MetadataOptions`— Konfigurasi layanan metadata instance untuk instance. Untuk informasi selengkapnya, lihat Menggunakan Layanan Metadata Instance untuk mengakses metadata instance.

Untuk jenis Armada EC2`instant`, Anda dapat menentukan parameter Manajer Sistem alih-alih ID AMI. Anda dapat menentukan parameter Manajer Sistem di override atau di template peluncuran. Untuk informasi selengkapnya, lihat Gunakan parameter Systems Manager alih-alih ID AMI.

Untuk jenis Armada EC2`instant`, Anda juga dapat menentukan data pengguna pada tingkat spesifikasi template peluncuran menggunakan parameter. `LaunchTemplateSpecificationUserData` Dengan parameter ini, Anda dapat memberikan data pengguna yang dikodekan base64 tanpa memasukkannya ke dalam template peluncuran itu sendiri. Data pengguna tidak boleh melebihi 16 KB sebelum pengkodean base64.

Anda dapat menentukan parameter armada dalam file JSON. Untuk informasi tentang semua parameter yang mungkin dapat Anda tentukan, lihatLihat semua opsi konfigurasi Armada EC2.

Untuk contoh konfigurasi armada, lihatContoh konfigurasi CLI untuk Armada EC2.

Saat ini tidak ada dukungan konsol untuk membuat Armada EC2.

Untuk membuat Armada EC2

Gunakan perintah create-fleet untuk membuat armada dan menentukan file JSON yang berisi parameter konfigurasi armada.

```
aws ec2 create-fleet --cli-input-json file://file_name.json
```

Berikut adalah contoh output untuk armada tipe `request` atau `maintain`.

```
{
    "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE"
```

```
}
```

Berikut adalah contoh output untuk tipe armada `instant` yang meluncurkan kapasitas target.

Respons mencakup ID Zona Ketersediaan, nama Zona Ketersediaan, dan ID subnet untuk setiap set instans yang diluncurkan.

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [],
  "Instances": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c5.large",
          "AvailabilityZone": "us-east-1a"
        }
      },
      "Lifecycle": "on-demand",
      "InstanceIds": [
        "i-1234567890abcdef0",
        "i-9876543210abcdef9"
      ],
      "InstanceType": "c5.large",
      "Platform": null,
      "AvailabilityZoneId": "use1-az1",
      "AvailabilityZone": "us-east-1a",
      "SubnetId": "subnet-0123456789abcdefEXAMPLE"
    },
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c4.large",
          "AvailabilityZone": "us-east-1a"
        }
```

```
      },
      "Lifecycle": "on-demand",
      "InstanceIds": [
        "i-5678901234abcdef0",
        "i-5432109876abcdef9"
      ],
      "AvailabilityZoneId": "use1-az1",
      "AvailabilityZone": "us-east-1a",
      "SubnetId": "subnet-0123456789abcdefEXAMPLE"
    }
  ]
}
```

Berikut adalah contoh output untuk armada tipe `instant` yang meluncurkan sebagian kapasitas target dengan kesalahan untuk instans yang tidak diluncurkan.

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c4.xlarge",
          "AvailabilityZone": "us-east-1a",
        }
      },
      "Lifecycle": "on-demand",
      "ErrorCode": "InsufficientInstanceCapacity",
      "ErrorMessage": ""
    },
  ],
  "Instances": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
```

```
          "InstanceType": "c5.large",
          "AvailabilityZone": "us-east-1a"
        }
      },
      "Lifecycle": "on-demand",
      "InstanceIds": [
        "i-1234567890abcdef0",
        "i-9876543210abcdef9"
      ]
  ]
}
```

Berikut adalah contoh output untuk armada tipe `instant` yang tidak meluncurkan instans.

```
{
  "FleetId": "fleet-12a34b55-67cd-8ef9-ba9b-9208dEXAMPLE",
  "Errors": [
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c4.xlarge",
          "AvailabilityZone": "us-east-1a",
        }
      },
      "Lifecycle": "on-demand",
      "ErrorCode": "InsufficientCapacity",
      "ErrorMessage": ""
    },
    {
      "LaunchTemplateAndOverrides": {
        "LaunchTemplateSpecification": {
          "LaunchTemplateId": "lt-01234a567b8910abcEXAMPLE",
          "Version": "1"
        },
        "Overrides": {
          "InstanceType": "c5.large",
          "AvailabilityZone": "us-east-1a",
        }
      },
```

```
      "Lifecycle": "on-demand",
      "ErrorCode": "InsufficientCapacity",
      "ErrorMessage": ""
    },
  ],
  "Instances": []
}
```

## Membuat Armada EC2 yang menggantikan Instans Spot yang tidak sehat

Armada EC2 memeriksa status kondisi instans di armada setiap dua menit. Status kondisi instans adalah `healthy` atau `unhealthy`.

Armada EC2 menentukan status kondisi instans dengan menggunakan pemeriksaan status yang disediakan oleh Amazon EC2. Sebuah instans ditentukan sebagai `unhealthy` jika status pemeriksaan status instans atau pemeriksaan status sistemnya `impaired` dalam tiga kali pemeriksaan kondisi secara berturut-turut. Untuk informasi selengkapnya, lihat Pemeriksaan status untuk instans Amazon EC2.

Anda dapat mengonfigurasi armada untuk mengganti Instans Spot yang tidak sehat. Setelah mengatur `ReplaceUnhealthyInstances` ke `true`, Instans Spot diganti ketika dilaporkan sebagai `unhealthy`. Armada tersebut dapat berada di bawah kapasitas targetnya selama beberapa menit saat Instans Spot yang tidak sehat sedang diganti.

Persyaratan

- Penggantian pemeriksaan kondisi hanya didukung untuk Armada EC2 yang mempertahankan kapasitas target (armada tipe `maintain`), dan bukan untuk armada tipe `request` atau `instant`.
- Penggantian pemeriksaan kondisi hanya didukung untuk Instans Spot. Fitur ini tidak didukung untuk On-Demand Instans.
- Anda dapat mengonfigurasi Armada EC2 untuk mengganti instans yang tidak sehat hanya saat Anda membuatnya.
- Pengguna dapat menggunakan penggantian pemeriksaan kondisi hanya jika memiliki izin untuk memanggil tindakan `ec2:DescribeInstanceStatus`.

Untuk mengonfigurasi Armada EC2 guna mengganti Instans Spot yang tidak sehat

1. Gunakan informasi untuk membuat Armada EC2 diMembuat Armada EC2.

2. Untuk mengonfigurasi armada untuk menggantikan Instans Spot yang tidak sehat, dalam file JSON, for`ReplaceUnhealthyInstances`, tentukan`true`.

## Lihat semua opsi konfigurasi Armada EC2

Untuk melihat daftar lengkap parameter konfigurasi Armada EC2, Anda dapat membuat file JSON. Untuk deskripsi setiap parameter, lihat create-fleet.

Untuk menghasilkan file JSON dengan semua parameter Armada EC2 yang mungkin

Gunakan perintah create-fleet (AWS CLI) dan parameter `--generate-cli-skeleton` untuk membuat file JSON Armada EC2, dan arahkan output ke file untuk menyimpannya.

```
aws ec2 create-fleet \
    --generate-cli-skeleton input > ec2createfleet.json
```

Berikut ini adalah output contoh.

```
{
    "DryRun": true,
    "ClientToken": "",
    "SpotOptions": {
        "AllocationStrategy": "price-capacity-optimized",
        "MaintenanceStrategies": {
            "CapacityRebalance": {
                "ReplacementStrategy": "launch"
            }
        },
        "InstanceInterruptionBehavior": "hibernate",
        "InstancePoolsToUseCount": 0,
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true,
        "MinTargetCapacity": 0,
        "MaxTotalPrice": ""
    },
    "OnDemandOptions": {
        "AllocationStrategy": "prioritized",
        "CapacityReservationOptions": {
            "UsageStrategy": "use-capacity-reservations-first"
        },
        "SingleInstanceType": true,
        "SingleAvailabilityZone": true,
```

```
        "MinTargetCapacity": 0,
        "MaxTotalPrice": ""
    },
    "ExcessCapacityTerminationPolicy": "termination",
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "",
                "LaunchTemplateName": "",
                "Version": ""
            },
            "Overrides": [
                {
                    "InstanceType": "r5.metal",
                    "MaxPrice": "",
                    "SubnetId": "",
                    "AvailabilityZone": "",
                    "WeightedCapacity": 0.0,
                    "Priority": 0.0,
                    "Placement": {
                        "AvailabilityZone": "",
                        "Affinity": "",
                        "GroupName": "",
                        "PartitionNumber": 0,
                        "HostId": "",
                        "Tenancy": "dedicated",
                        "SpreadDomain": "",
                        "HostResourceGroupArn": ""
                    },
                    "InstanceRequirements": {
                        "VCpuCount": {
                            "Min": 0,
                            "Max": 0
                        },
                        "MemoryMiB": {
                            "Min": 0,
                            "Max": 0
                        },
                        "CpuManufacturers": [
                            "amd"
                        ],
                        "MemoryGiBPerVCpu": {
                            "Min": 0.0,
                            "Max": 0.0
```

```
                    },
                    "ExcludedInstanceTypes": [
                        ""
                    ],
                    "InstanceGenerations": [
                        "previous"
                    ],
                    "SpotMaxPricePercentageOverLowestPrice": 0,
                    "OnDemandMaxPricePercentageOverLowestPrice": 0,
                    "BareMetal": "included",
                    "BurstablePerformance": "required",
                    "RequireHibernateSupport": true,
                    "NetworkInterfaceCount": {
                        "Min": 0,
                        "Max": 0
                    },
                    "LocalStorage": "excluded",
                    "LocalStorageTypes": [
                        "ssd"
                    ],
                    "TotalLocalStorageGB": {
                        "Min": 0.0,
                        "Max": 0.0
                    },
                    "BaselineEbsBandwidthMbps": {
                        "Min": 0,
                        "Max": 0
                    },
                    "AcceleratorTypes": [
                        "inference"
                    ],
                    "AcceleratorCount": {
                        "Min": 0,
                        "Max": 0
                    },
                    "AcceleratorManufacturers": [
                        "amd"
                    ],
                    "AcceleratorNames": [
                        "a100"
                    ],
                    "AcceleratorTotalMemoryMiB": {
                        "Min": 0,
                        "Max": 0
```

```
                        }
                    }
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 0,
        "OnDemandTargetCapacity": 0,
        "SpotTargetCapacity": 0,
        "DefaultTargetCapacityType": "on-demand",
        "TargetCapacityUnitType": "memory-mib"
    },
    "TerminateInstancesWithExpiration": true,
    "Type": "instant",
    "ValidFrom": "1970-01-01T00:00:00",
    "ValidUntil": "1970-01-01T00:00:00",
    "ReplaceUnhealthyInstances": true,
    "TagSpecifications": [
        {
            "ResourceType": "fleet",
            "Tags": [
                {
                    "Key": "",
                    "Value": ""
                }
            ]
        }
    ],
    "Context": ""
}
```

## Menandai permintaan Armada EC2 yang baru atau yang sudah ada serta instans serta volume yang diluncurkan

Untuk membantu mengkategorikan dan mengelola permintaan EC2 Fleet serta instans serta volume yang diluncurkan, Anda dapat menandainya dengan metadata khusus. Anda dapat menetapkan tanda untuk permintaan Armada EC2 saat membuatnya, atau sesudahnya. Demikian pula, Anda dapat menetapkan tag ke instance dan volume saat diluncurkan oleh armada, atau sesudahnya.

Saat Anda menandai permintaan armada, instans dan volume yang diluncurkan oleh armada tidak ditandai secara otomatis. Anda perlu menandai instans dan volume yang diluncurkan oleh armada

secara eksplisit. Anda dapat memilih untuk menetapkan tag hanya untuk permintaan armada, atau hanya instans yang diluncurkan oleh armada, atau hanya volume yang dilampirkan ke instans yang diluncurkan oleh armada, atau hanya untuk antarmuka jaringan yang dibuat oleh armada, atau ke semuanya.

> **Note**
>
> Untuk jenis `instant` armada, Anda dapat menandai volume yang dilampirkan ke On-Demand Instans dan Instans Spot. Anda juga dapat menandai antarmuka jaringan untuk jenis `instant` armada. Untuk jenis `request` atau `maintain` armada, Anda hanya dapat menandai volume yang dilampirkan ke On-Demand Instans.

Untuk informasi selengkapnya tentang cara kerja tag, lihat Tandai sumber daya Amazon EC2 Anda.

Prasyarat

Berikan izin kepada pengguna untuk menandai sumber daya. Untuk informasi selengkapnya, lihat Contoh: Memberi tanda pada sumber daya.

Berikan izin kepada pengguna untuk menandai sumber daya

Buat kebijakan IAM yang mencakup hal-hal berikut:

- Tindakan `ec2:CreateTags`. Tindakan ini memberikan izin kepada pengguna untuk membuat tanda.
- Tindakan `ec2:CreateFleet`. Tindakan ini memberikan izin kepada pengguna untuk membuat permintaan Armada EC2.
- Untuk `Resource`, kami sarankan Anda menentukan `"*"`. Tindakan ini memungkinkan pengguna untuk menandai semua tipe sumber daya.

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.

- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti instruksi dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

Untuk menandai permintaan Armada EC2 baru

Untuk menandai permintaan Armada EC2 saat Anda membuatnya, tentukan pasangan kunci-nilai di file JSON yang digunakan untuk membuat armada. Nilai untuk `ResourceType` harus `fleet`. Jika Anda menentukan nilai lain, permintaan armada akan gagal.

Untuk menandai instans dan volume yang diluncurkan oleh Armada EC2

Untuk menandai instans dan volume saat diluncurkan oleh armada tersebut, tentukan tanda di luncurkan templat yang direferensikan dalam permintaan Armada EC2.

> **Note**
>
> Anda tidak dapat menandai volume yang dilampirkan ke Instans Spot yang diluncurkan oleh tipe armada `request` atau `maintain`.

Untuk menandai permintaan, instans, dan volume EC2 Fleet yang ada

Gunakan perintah create-tags untuk menandai sumber daya yang ada.

```
aws ec2 create-tags \
    --resources fleet-12a34b55-67cd-8ef9-
ba9b-9208dEXAMPLE i-1234567890abcdef0 vol-1234567890EXAMPLE \
    --tags Key=purpose,Value=test
```

# Jelaskan Armada EC2, instansnya, dan kejadiannya

Anda dapat menjelaskan konfigurasi Armada EC2, instans di Armada EC2, dan riwayat peristiwa Armada EC2.

Tugas

- Jelaskan Armada EC2 Anda
- Jelaskan semua instans di Armada EC2 Anda
- Jelaskan riwayat acara untuk Armada EC2 Anda

## Jelaskan Armada EC2 Anda

AWS CLI

Untuk menggambarkan Armada EC2 Anda

Gunakan perintah describe-fleets.

```
aws ec2 describe-fleets \
    --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Berikut ini adalah output contoh.

```
{
    "Fleets": [
        {
            "ActivityStatus": "fulfilled",
            "CreateTime": "2022-02-09T03:35:52+00:00",
            "FleetId": "fleet-364457cd-3a7a-4ed9-83d0-7b63e51bb1b7",
            "FleetState": "active",
            "ExcessCapacityTerminationPolicy": "termination",
            "FulfilledCapacity": 2.0,
            "FulfilledOnDemandCapacity": 0.0,
            "LaunchTemplateConfigs": [
                {
                    "LaunchTemplateSpecification": {
                        "LaunchTemplateName": "my-launch-template",
                        "Version": "$Latest"
                    }
                }
            ],
```

```
            "TargetCapacitySpecification": {
                "TotalTargetCapacity": 2,
                "OnDemandTargetCapacity": 0,
                "SpotTargetCapacity": 2,
                "DefaultTargetCapacityType": "spot"
            },
            "TerminateInstancesWithExpiration": false,
            "Type": "maintain",
            "ReplaceUnhealthyInstances": false,
            "SpotOptions": {
                "AllocationStrategy": "capacity-optimized",
                "InstanceInterruptionBehavior": "terminate"
            },
            "OnDemandOptions": {
                "AllocationStrategy": "lowestPrice"
            }
        }
    ]
}
```

PowerShell

Untuk menggambarkan Armada EC2 Anda

Gunakan Get-EC2FleetList cmdlet.

```
Get-EC2FleetList `
    -FleetId fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

## Jelaskan semua instans di Armada EC2 Anda

Daftar instans yang sedang berjalan yang dikembalikan diperbarui secara berkala dan mungkin sudah lawas.

AWS CLI

Untuk menjelaskan instance untuk Armada EC2 yang ditentukan

Gunakan perintah describe-fleet-instance.

```
aws ec2 describe-fleet-instances \
    --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Berikut ini adalah output contoh.

```
{
    "ActiveInstances": [
        {
            "InstanceId": "i-09cd595998cb3765e",
            "InstanceHealth": "healthy",
            "InstanceType": "m4.large",
            "SpotInstanceRequestId": "sir-86k84j6p"
        },
        {
            "InstanceId": "i-09cf95167ca219f17",
            "InstanceHealth": "healthy",
            "InstanceType": "m4.large",
            "SpotInstanceRequestId": "sir-dvxi7fsm"
        }
    ],
    "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
}
```

PowerShell

Untuk menjelaskan instance untuk Armada EC2 yang ditentukan

Gunakan Get-EC2FleetInstanceList cmdlet.

```
Get-EC2FleetInstanceList `
    -FleetId fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

## Jelaskan riwayat acara untuk Armada EC2 Anda

Untuk informasi selengkapnya tentang peristiwa dalam riwayat acara, lihatJenis acara Armada EC2.

AWS CLI

Untuk menjelaskan peristiwa untuk Armada EC2 yang ditentukan

Gunakan perintah describe-fleet-history.

```
aws ec2 describe-fleet-history \
    --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --start-time 2020-06-01T00:00:00Z
```

Berikut ini adalah output contoh.


```
{
    "HistoryRecords": [
        {
            "EventInformation": {
                "EventSubType": "submitted"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:05.000Z"
        },
        {
            "EventInformation": {
                "EventSubType": "active"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:15.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "t2.small, ami-07c8bc5c1ce9598c3, ...",
                "EventSubType": "progress"
            },
            "EventType": "fleetRequestChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
                "EventSubType": "launched",
                "InstanceId": "i-083a1c446e66085d2"
            },
            "EventType": "instanceChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        },
        {
            "EventInformation": {
                "EventDescription": "{\"instanceType\":\"t2.small\", ...}",
                "EventSubType": "launched",
                "InstanceId": "i-090db02406cc3c2d6"
            },
            "EventType": "instanceChange",
            "Timestamp": "2020-09-01T18:26:17.000Z"
        }
```

```
    ],
    "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
    "LastEvaluatedTime": "1970-01-01T00:00:00.000Z",
    "StartTime": "2020-06-01T00:00:00.000Z"
}
```

PowerShell

Untuk menjelaskan peristiwa untuk Armada EC2 yang ditentukan

Gunakan Get-EC2FleetHistory cmdlet.

```
Get-EC2FleetHistory `
    -FleetId fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE `
    -UtcStartTime 2020-06-01T00:00:00Z
```

## Memodifikasi Armada EC2

Anda dapat mengubah total kapasitas target, kapasitas Spot, dan On-Demand kapasitas Armada EC2. Anda juga dapat mengubah apakah instance yang sedang berjalan harus dihentikan jika kapasitas target total baru dikurangi di bawah ukuran armada saat ini.

### Pertimbangan-pertimbangan

Pertimbangkan hal berikut saat memodifikasi Armada EC2:

- Jenis armada — Anda hanya dapat memodifikasi jenis `maintain` Armada EC2. Anda tidak dapat memodifikasi Armada EC2 tipe `request` atau`instant`.
- Parameter armada — Anda dapat memodifikasi parameter berikut dari Armada EC2:
  - `target-capacity-specification`— Meningkatkan atau mengurangi kapasitas target untuk:
    - `TotalTargetCapacity`
    - `OnDemandTargetCapacity`
    - `SpotTargetCapacity`
  - `excess-capacity-termination-policy` – Apakah instans yang sedang berjalan harus diakhiri jika total kapasitas target Armada EC2 turun di bawah ukuran armada saat ini. Nilai yang valid adalah:

  - `no-termination`
  - `termination`
- Perilaku armada saat meningkatkan kapasitas target total — Saat Anda meningkatkan kapasitas target total, Armada EC2 meluncurkan instans tambahan sesuai dengan opsi pembelian instans yang ditentukan`DefaultTargetCapacityType`, yaitu Instans atau On-Demand Instans Spot, dan sesuai dengan strategi ??? alokasi yang ditentukan.
- Perilaku armada saat mengurangi kapasitas target Spot — Saat Anda mengurangi kapasitas target Spot, Armada EC2 menghapus permintaan terbuka yang melebihi kapasitas target baru. Anda dapat meminta armada menghentikan Instans Spot hingga ukuran armada mencapai kapasitas target baru. Saat Armada EC2 mengakhiri Instans Spot karena kapasitas target berkurang, instans tersebut akan menerima pemberitahuan gangguan Instans Spot.

  Instans dipilih untuk penghentian berdasarkan strategi alokasi:

  - `capacity-optimized`— Menghentikan instance dari kumpulan dengan kapasitas paling sedikit yang tersedia.
  - `price-capacity-optimized`Menggunakan kombinasi harga dan kapasitas yang tersedia: mengakhiri instance dari kumpulan dengan kapasitas paling sedikit yang tersedia dan yang merupakan harga tertinggi di antara kumpulan ini.
  - `diversified`— Menghentikan instance di semua kumpulan.
  - `lowest-price`— Menghentikan instans dari kumpulan dengan harga tertinggi.

  Atau, Anda dapat meminta agar Armada EC2 mempertahankan armada pada ukuran saat ini, tetapi tidak mengganti Instans Spot apa pun yang terganggu atau yang Anda akhiri secara manual.
- Status Armada — Anda dapat memodifikasi Armada EC2 yang berada di `active` negara bagian `submitted` atau. Saat Anda memodifikasi armada, maka armada tersebut memasuki status `modifying`.

## Perintah untuk memodifikasi Armada EC2

AWS CLI

Untuk memodifikasi total kapasitas target Armada EC2

Gunakan perintah modify-fleet.

```
aws ec2 modify-fleet \
```

```
    --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity-specification TotalTargetCapacity=20
```

Jika Anda mengurangi kapasitas target tetapi ingin mempertahankan armada pada ukuran saat ini, Anda dapat memodifikasi contoh sebelumnya sebagai berikut.

```
aws ec2 modify-fleet \
    --fleet-id fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity-specification TotalTargetCapacity=10 \
    --excess-capacity-termination-policy no-termination
```

PowerShell

Untuk memodifikasi total kapasitas target Armada EC2

Gunakan Edit-EC2Fleet cmdlet.

```
Edit-EC2Fleet `
    -FleetId "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TargetCapacitySpecification_TotalTargetCapacity 20
```

Jika Anda mengurangi kapasitas target tetapi ingin mempertahankan armada pada ukuran saat ini, Anda dapat memodifikasi contoh sebelumnya sebagai berikut.

```
Edit-EC2Fleet `
    -FleetId "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TargetCapacitySpecification_TotalTargetCapacity 20 `
    -ExcessCapacityTerminationPolicy "NoTermination"
```

## Menghapus permintaan Armada EC2 dan instans di armada

Jika Anda tidak lagi memerlukan permintaan EC2 Fleet, Anda dapat menghapusnya. Setelah Anda menghapus permintaan armada, semua permintaan Spot yang terkait dengan armada dibatalkan, sehingga tidak ada Instans Spot baru yang diluncurkan.

Saat menghapus permintaan EC2 Fleet, Anda juga harus menentukan apakah Anda ingin menghentikan semua instansnya. Ini termasuk On-Demand Instans dan Instans Spot. Untuk `instant` armada, Armada EC2 harus menghentikan instans saat armada dihapus. Armada `instant` yang dihapus dengan instans yang sedang berjalan tidak didukung.

> **Warning**
>
> Mengakhiri instance bersifat permanen dan tidak dapat diubah.
> Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
> Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Jika Anda menentukan bahwa instance harus dihentikan saat permintaan armada dihapus, permintaan armada memasuki `deleted_terminating` status. Jika tidak, armada masuk ke status `deleted_running` dan instans terus berjalan hingga diinterupsi atau Anda mengakhirinya secara manual.

Pembatasan

- Anda dapat menghapus hingga 25 armada jenis `instant` dalam satu operasi.
- Anda dapat menghapus hingga 100 armada jenis `maintain` atau `request` dalam satu operasi.
- Anda dapat menghapus hingga 125 armada dalam satu operasi, asalkan Anda tidak melebihi kuota untuk setiap jenis armada, seperti yang ditentukan di atas.
- Jika Anda melebihi jumlah armada yang ditentukan untuk dihapus, tidak ada armada yang dihapus.
- Armada `instant` yang dihapus dengan instans yang sedang berjalan tidak didukung. Saat Anda menghapus `instant` armada, Amazon EC2 secara otomatis mengakhiri semua instansnya. Untuk `instant` armada dengan lebih dari 1000 instans, permintaan penghapusan mungkin gagal. Jika armada Anda memiliki lebih dari 1000 instans, pertama-tama hentikan sebagian besar instans secara manual, sisakan 1000 atau kurang. Kemudian hapus armada, dan instans yang tersisa akan dihentikan secara otomatis.

AWS CLI

Untuk menghapus permintaan EC2 Fleet dan menghentikan instansnya

Gunakan perintah delete-fleets dengan opsi. `--terminate-instances`

```
aws ec2 delete-fleets \
    --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --terminate-instances
```

Berikut ini adalah output contoh.

```
{
    "UnsuccessfulFleetDeletions": [],
    "SuccessfulFleetDeletions": [
        {
            "CurrentFleetState": "deleted_terminating",
            "PreviousFleetState": "active",
            "FleetId": "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
        }
    ]
}
```

Untuk menghapus permintaan Armada EC2 tanpa mengakhiri instansnya

Ubah contoh sebelumnya dengan menggunakan `--no-terminate-instances` opsi sebagai gantinya. Perhatikan bahwa `--no-terminate-instances` tidak didukung untuk arm `instant` ada.

```
aws ec2 delete-fleets \
    --fleet-ids fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --no-terminate-instances
```

Berikut ini adalah output contoh.

```
{
    "UnsuccessfulFleetDeletions": [],
    "SuccessfulFleetDeletions": [
        {
            "CurrentFleetState": "deleted_running",
            "PreviousFleetState": "active",
            "FleetId": "fleet-4b8aaae8-dfb5-436d-a4c6-3dafa4c6b7dcEXAMPLE"
        }
    ]
}
```

PowerShell

Untuk menghapus permintaan EC2 Fleet dan menghentikan instansnya

Gunakan Remove-EC2Fleet cmdlet dengan parameter. `-TerminateInstance`

```
Remove-EC2Fleet `
    -FleetId "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TerminateInstance $true
```

Untuk menghapus permintaan Armada EC2 tanpa mengakhiri instansnya

Ubah contoh sebelumnya dengan mengubah nilai `-TerminateInstance` parameter.

```
Remove-EC2Fleet `
    -FleetId "fleet-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TerminateInstance $false
```

## Memecahkan masalah saat armada gagal dihapus

Jika permintaan Armada EC2 gagal dihapus, `UnsuccessfulFleetDeletions` dalam output mengembalikan ID permintaan Armada EC2, kode kesalahan, dan pesan kesalahan.

Kode kesalahannya adalah:

- `ExceededInstantFleetNumForDeletion`
- `fleetIdDoesNotExist`
- `fleetIdMalformed`
- `fleetNotInDeletableState`
- `NoTerminateInstancesNotSupported`
- `UnauthorizedOperation`
- `unexpectedError`

### Pecahkan masalah **`ExceededInstantFleetNumForDeletion`**

Jika Anda mencoba menghapus lebih dari 25 armada `instant` dalam satu permintaan, kesalahan `ExceededInstantFleetNumForDeletion` akan dikembalikan. Berikut adalah contoh output untuk kesalahan ini.

```
{
    "UnsuccessfulFleetDeletions": [
     {
          "FleetId": " fleet-5d130460-0c26-bfd9-2c32-0100a098f625",
          "Error": {
                  "Message": "Can’t delete more than 25 instant fleets in a single
 request.",
                  "Code": "ExceededInstantFleetNumForDeletion"
           }
     },
     {
           "FleetId": "fleet-9a941b23-0286-5bf4-2430-03a029a07e31",
           "Error": {
                  "Message": "Can’t delete more than 25 instant fleets in a single
 request.",
                  "Code": "ExceededInstantFleetNumForDeletion"
            }
     }
     .
     .
     .
     ],
     "SuccessfulFleetDeletions": []
}
```

### Pecahkan masalah **NoTerminateInstancesNotSupported**

Jika Anda menentukan bahwa instans dalam armada `instant` tidak boleh diakhiri saat menghapus armada, kesalahan `NoTerminateInstancesNotSupported` akan dikembalikan. `--no-terminate-instances` tidak didukung untuk armada `instant`. Berikut adalah contoh output untuk kesalahan ini.

```
{
      "UnsuccessfulFleetDeletions": [
            {
                  "FleetId": "fleet-5d130460-0c26-bfd9-2c32-0100a098f625",
                  "Error": {
                         "Message": "NoTerminateInstances option is not supported for
 instant fleet",
                         "Code": "NoTerminateInstancesNotSupported"
                   }
            }
```

```
        ],
        "SuccessfulFleetDeletions": []
}
```

### Pecahkan masalah **UnauthorizedOperation**

Jika Anda tidak memiliki izin untuk mengakhiri instans, Anda akan mendapatkan kesalahan `UnauthorizedOperation` saat menghapus armada yang harus mengakhiri instansnya. Berikut ini adalah respons kesalahannya.

```
<Response><Errors><Error><Code>UnauthorizedOperation</Code><Message>You are not
 authorized to perform this
operation. Encoded authorization failure message: VvuncIxj7Z_CPGNYXWqnuFV-
YjByeAU66Q9752NtQ-I3-qnDLWs6JLFd
KnSMMiq5s6cGqjjPtEDpsnGHzzyHasFHOaRYJpaDVravoW25azn6KNkUQQlFwhJyujt2dtNCdduJfrqcFYAjlEiRMkfDHt7
BHturzDK6A560Y2nDSUiMmAB1y9UNtqaZJ9SNe5sNxKMqZaqKtjRbk02RZu5V2vn9VMk6fm2aMVHbY9JhLvGypLcMUjtJ76
VPiU5v2s-
UgZ7h0p2yth6ysUdhlONg6dBYu8_y_HtEI54invCj4CoK0qawqzMNe6rcmCQHvtCxtXsbkgyaEbcwmrm2m01-
EMhekLFZeJLr
DtYOpYcEl4_nWFX1wtQDCnNNCmxnJZAoJvb3VMDYpDTsxjQv1PxODZuqWHs23YXWVywzgnLtHeRf2o4lUhGBw17mXsS07k7
PT9vrHtQiILor5VVTsjSPWg7edj__1rsnXhwPSu8gI48ZLRGrPQqFq0RmKO_QIE8N8s6NWzCK4yoX-9gDcheurOGpkprPIC
</Message></Error></Errors><RequestID>89b1215c-7814-40ae-a8db-41761f43f2b0</
RequestID></Response>
```

Untuk mengatasi kesalahan tersebut, Anda harus menambahkan tindakan `ec2:TerminateInstances` ke kebijakan IAM, seperti yang ditunjukkan pada contoh berikut.

## Bekerja dengan Spot Fleet

> **Important**
>
> Kami sangat tidak menyarankan menggunakan Spot Fleet karena menggunakan API lama tanpa investasi yang direncanakan. Jika Anda ingin mengelola siklus hidup instans, gunakan EC2 Fleet sebagai gantinya. Jika Anda tidak ingin mengelola siklus hidup instans, gunakan grup Penskalaan Otomatis sebagai gantinya. Gunakan Spot Fleet hanya jika Anda memerlukan dukungan konsol untuk kasus penggunaan di mana Anda akan menggunakan EC2 Fleet. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Untuk mulai menggunakan Spot Fleet, buat permintaan yang menyertakan total kapasitas target untuk Instans Spot, On-Demand bagian opsional, dan tentukan AMI dan pasangan kunci secara manual, atau tentukan template peluncuran yang menyertakan konfigurasi untuk instance dalam armada. Anda dapat secara opsional menentukan parameter tambahan, atau membiarkan armada menggunakan nilai default. Anda juga dapat menandai permintaan armada, serta instance serta volumenya, saat membuat armada.

Armada akan meluncurkan On-Demand Instans jika ada kapasitas yang tersedia, dan meluncurkan Instans Spot saat harga maksimum melebihi harga Spot dan kapasitas yang tersedia.

Setelah armada Anda diluncurkan, Anda dapat menjelaskan permintaan armada, instance dalam armada, dan peristiwa armada apa pun. Anda juga dapat menetapkan tag tambahan sesuai kebutuhan.

Jika Anda perlu mengubah parameter armada apa pun, seperti kapasitas target total, Anda dapat memodifikasi armada, asalkan dikonfigurasi untuk mempertahankan kapasitas. Anda tidak dapat mengubah kapasitas permintaan satu kali setelah dikirimkan.

Permintaan armada tetap aktif hingga kedaluwarsa atau Anda membatalkannya (menghapus). Saat Anda membatalkan permintaan armada, Anda dapat menghentikan instans atau membiarkannya berjalan. Jika Anda memilih untuk membiarkannya berjalan, On-Demand Instans akan berjalan sampai Anda menghentikannya, dan Instans Spot berjalan hingga terganggu atau Anda menghentikannya.

Topik


- Status permintaan Armada Spot
- Izin Armada Spot
- Membuat Armada Spot
- Menandai permintaan Spot Fleet baru atau yang sudah ada serta instans serta volume yang diluncurkan
- Menjelaskan permintaan Spot Fleet, instansnya, dan riwayat peristiwa
- Memodifikasi permintaan Armada Spot
- Membatalkan (menghapus) permintaan Spot Fleet
- Memahami penskalaan otomatis untuk Spot Fleet

# Status permintaan Armada Spot

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Permintaan Spot Fleet dapat berupa salah satu dari berbagai status, dengan setiap status menunjukkan tahap yang berbeda dari siklus hidup permintaan dan pengelolaan instansnya.

Permintaan Armada Spot dapat berada dalam salah satu kondisi berikut:

`submitted`

Permintaan Armada Spot sedang dievaluasi dan Amazon EC2 sedang bersiap untuk meluncurkan jumlah target instans. Jika permintaan melebihi kuota Spot Fleet Anda, permintaan tersebut akan segera dibatalkan.

`active`

Armada Spot telah divalidasi dan Amazon EC2 berusaha mempertahankan jumlah target Instans Spot yang sedang berjalan. Permintaan tetap dalam keadaan ini sampai dimodifikasi atau dibatalkan.

`modifying`

Permintaan Armada Spot sedang dimodifikasi. Permintaan tetap dalam keadaan ini sampai modifikasi diproses sepenuhnya atau permintaan dibatalkan. Hanya tipe armada `maintain` yang dapat dimodifikasi. Status ini tidak berlaku untuk jenis `request` armada satu kali.

`cancelled_running`

Armada Spot dibatalkan (dihapus) dan tidak meluncurkan Instans Spot tambahan. Instans yang ada terus berjalan sampai diinterupsi atau diakhiri secara manual. Permintaan tetap berada dalam status ini sampai semua instans diinterupsi atau diakhiri.

`cancelled_terminating`

Armada Spot dibatalkan (dihapus) dan instansnya dihentikan. Permintaan tetap berada dalam status ini sampai semua instans diakhiri.

`cancelled`

Armada Spot dibatalkan (dihapus) dan tidak memiliki instans yang berjalan. Permintaan tersebut dihapus dua hari setelah instansnya diakhiri.

## Izin Armada Spot

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Jika pengguna Anda akan membuat atau mengelola Armada Spot, Anda perlu memberinya izin yang diperlukan.

Jika Anda menggunakan konsol Amazon EC2 untuk membuat Armada Spot, konsol ini akan membuat dua peran tertaut layanan bernama `AWSServiceRoleForEC2SpotFleet` dan `AWSServiceRoleForEC2Spot`, serta peran bernama `aws-ec2-spot-fleet-tagging-role` yang memberikan izin kepada Armada Spot untuk meminta, meluncurkan, mengakhiri, dan menandai sumber daya atas nama Anda. Jika Anda menggunakan AWS CLI atau API, Anda harus memastikan bahwa peran ini ada.

Gunakan petunjuk berikut untuk memberikan izin yang diperlukan dan membuat peran.

Izin dan peran

- Memberikan izin kepada pengguna untuk Armada Spot
- Service-linked peran untuk Spot Fleet
- Service-linked peran untuk Instans Spot
- Peran IAM untuk menandai Armada Spot

### Memberikan izin kepada pengguna untuk Armada Spot

Jika pengguna Anda akan membuat atau mengelola Armada Spot, pastikan untuk memberinya izin yang diperlukan.

Untuk membuat kebijakan Armada Spot

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Kebijakan, Buat kebijakan.
3. Di halaman Buat kebijakan, pilih JSON, dan ganti teks dengan yang berikut ini.

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances",
                "ec2:CreateTags",
                "ec2:RequestSpotFleet",
                "ec2:ModifySpotFleetRequest",
                "ec2:CancelSpotFleetRequests",
                "ec2:DescribeSpotFleetRequests",
                "ec2:DescribeSpotFleetInstances",
                "ec2:DescribeSpotFleetRequestHistory"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::*:role/aws-ec2-spot-fleet-tagging-role"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole",
                "iam:ListRoles",
                "iam:ListInstanceProfiles"
            ],
            "Resource": "*"
        }
    ]
}
```

Contoh kebijakan sebelumnya memberikan izin yang diperlukan kepada pengguna untuk sebagian besar kasus penggunaan Armada Spot. Untuk membatasi pengguna ke tindakan API tertentu, tentukan hanya tindakan API tersebut saja.

### API EC2 dan IAM yang diperlukan

API berikut harus disertakan dalam kebijakan:

- `ec2:RunInstances` – Diperlukan untuk meluncurkan instans di Armada Spot
- `ec2:CreateTags` – Diperlukan untuk menandai permintaan, instans, atau volume Armada Spot
- `iam:PassRole` – Diperlukan untuk menentukan peran Armada Spot
- `iam:CreateServiceLinkedRole` – Diperlukan untuk membuat peran tertaut-layanan
- `iam:ListRoles` – Diperlukan untuk melakukan enumerasi peran IAM yang ada
- `iam:ListInstanceProfiles` – Diperlukan untuk melakukan enumerasi profil instans yang sudah ada

> **Important**
>
> Jika Anda menentukan peran untuk profil instans IAM dalam spesifikasi peluncuran atau templat peluncuran, Anda harus memberikan izin kepada pengguna untuk meneruskan peran tersebut ke layanan. Untuk melakukan ini, dalam kebijakan IAM sertakan `"arn:aws:iam::*:role/IamInstanceProfile-role"` sebagai sumber daya untuk tindakan `iam:PassRole`. Untuk informasi selengkapnya, lihat Memberikan izin pengguna untuk meneruskan peran ke AWS layanan di Panduan Pengguna IAM.

### API Armada Spot

Tambahkan tindakan API Armada Spot berikut ke kebijakan Anda, jika diperlukan:

- `ec2:RequestSpotFleet`
- `ec2:ModifySpotFleetRequest`
- `ec2:CancelSpotFleetRequests`
- `ec2:DescribeSpotFleetRequests`

- `ec2:DescribeSpotFleetInstances`
- `ec2:DescribeSpotFleetRequestHistory`

API IAM opsional

(Opsional) Untuk memungkinkan pengguna membuat peran atau profil instans menggunakan konsol IAM, Anda juga harus menambahkan tindakan berikut ke kebijakan:

- `iam:AddRoleToInstanceProfile`
- `iam:AttachRolePolicy`
- `iam:CreateInstanceProfile`
- `iam:CreateRole`
- `iam:GetRole`
- `iam:ListPolicies`

4. Pilih Tinjau kebijakan.
5. Pada halaman Tinjau kebijakan, masukkan nama dan deskripsi kebijakan, dan pilih Buat kebijakan.
6. Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

   - Pengguna dan grup di AWS IAM Identity Center:

     Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .

   - Pengguna yang dikelola di IAM melalui penyedia identitas:

     Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.

   - Pengguna IAM:
     - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
     - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

## Service-linked peran untuk Spot Fleet

Amazon EC2 menggunakan peran tertaut layanan untuk izin yang diperlukan untuk memanggil layanan AWS lain atas nama Anda. Peran terkait layanan adalah jenis peran IAM unik yang ditautkan langsung ke layanan. AWS Service-linked peran menyediakan cara aman untuk mendelegasikan izin ke AWS layanan karena hanya layanan tertaut yang dapat mengambil peran terkait layanan. Untuk informasi selengkapnya, lihat Service-linked peran di Panduan Pengguna IAM.

Amazon EC2 menggunakan peran terkait layanan bernama AWSServiceRoleForEC2Spot Fleet untuk meluncurkan dan mengelola instans atas nama Anda.

> **Important**
>
> Jika Anda menentukan AMI terenkripsi atau snapshot Amazon EBS terenkripsi di Armada Spot Anda, Anda harus memberikan izin peran Ar AWSServiceRoleForEC2Spot mada untuk menggunakan CMK sehingga Amazon EC2 dapat meluncurkan instans atas nama Anda. Untuk informasi selengkapnya, lihat Berikan akses ke CMK untuk digunakan dengan AMI dan snapshot EBS terenkripsi.

### Izin yang diberikan oleh AWSServiceRoleForEC2SpotFleet

Peran Ar AWSServiceRoleForEC2Spot mada memberikan izin Armada Spot untuk meminta, meluncurkan, menghentikan, dan menandai instance atas nama Anda. Amazon EC2 menggunakan peran tertaut layanan ini untuk menyelesaikan tindakan berikut:

- `ec2:RequestSpotInstances` - Meminta Instans Spot
- `ec2:RunInstances` - Meluncurkan instans
- `ec2:TerminateInstances` - Mengakhiri instans
- `ec2:DescribeImages` - Mendeskripsikan Amazon Machine Image (AMI) untuk instans
- `ec2:DescribeInstanceStatus` - Mendeskripsikan status instans
- `ec2:DescribeSubnets` - Mendeskripsikan subnet untuk instans
- `ec2:CreateTags` - Menambahkan tanda ke permintaan, instans, dan volume Armada Spot
- `elasticloadbalancing:RegisterInstancesWithLoadBalancer` - Menambahkan instans yang ditentukan ke penyeimbang beban yang ditentukan
- `elasticloadbalancing:RegisterTargets` - Mendaftarkan target yang ditentukan dengan grup target yang ditentukan

### Membuat peran tertaut layanan

Dalam sebagian besar situasi, Anda tidak perlu membuat peran tertaut layanan secara manual. Amazon EC2 membuat peran terkait AWSServiceRoleForEC2Spot layanan Armada saat pertama kali Anda membuat Spot Fleet menggunakan konsol.

Jika Anda memiliki permintaan Spot Fleet aktif sebelum Oktober 2017, ketika Amazon EC2 mulai mendukung peran terkait layanan ini, Amazon EC2 membuat peran Ar AWSServiceRoleForEC2Spot mada di akun Anda. AWS Untuk informasi selengkapnya, lihat Peran baru muncul di AWS akun saya di Panduan Pengguna IAM.

Jika Anda menggunakan AWS CLI atau API untuk membuat Spot Fleet, Anda harus terlebih dahulu memastikan bahwa peran ini ada.

Untuk membuat AWSServiceRoleForEC2SpotFleet peran untuk Spot Fleet menggunakan konsol

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Pilih Buat peran.
4. Pada halaman Pilih entitas tepercaya, lakukan hal berikut:
   a. Untuk Ti pe entitas tepercaya, pilih AWS layanan.
   b. Di bawah Kasus penggunaan, untuk Layanan atau kasus penggunaan, pilih EC2.
   c. Untuk kasus penggunaan, pilih EC2 - Spot Fleet.

      > **Note**
      >
      > Kas us penggunaan EC2 - Spot Fleet akan secara otomatis membuat kebijakan dengan izin IAM yang diperlukan dan akan menyarankan AWSEC2SpotFleetServiceRolePolicy sebagai nama peran.

   d. Pilih Berikutnya.
5. Pada halaman Tambahkan izin, pilih Berikutnya.
6. Pada halaman Nama, ulasan, dan buat, pilih Buat peran.

Untuk membuat AWSServiceRoleForEC2SpotFleet peran untuk Spot Fleet menggunakan AWS CLI

Gunakan perintah create-service-linked-role sebagai berikut.

```
aws iam create-service-linked-role --aws-service-name spotfleet.amazonaws.com
```

Jika Anda tidak perlu lagi menggunakan Spot Fleet, sebaiknya hapus peran AWSServiceRoleForEC2Spot Armada. Setelah peran ini dihapus dari akun Anda, Amazon EC2 akan membuat peran lagi jika Anda meminta Armada Spot dengan menggunakan konsol. Untuk informasi selengkapnya, lihat Menghapus peran terkait layanan dalam Panduan Pengguna IAM.

Berikan akses ke CMK untuk digunakan dengan AMI dan snapshot EBS terenkripsi

Jika Anda menentukan AMI terenkripsi atau snapshot Amazon EBS terenkripsi dalam permintaan Spot Fleet Anda dan Anda menggunakan kunci yang dikelola pelanggan untuk enkripsi, Anda harus memberikan izin peran Ar AWSServiceRoleForEC2Spot mada untuk menggunakan CMK sehingga Amazon EC2 dapat meluncurkan instans atas nama Anda. Untuk melakukannya, Anda harus menambahkan pemberian izin ke CMK, seperti yang ditunjukkan dalam prosedur berikut.

Ketika memberikan izin, pemberian izin merupakan alternatif dari kebijakan kunci. Untuk informasi selengkapnya, lihat Menggunakan Pemberian Izin dan Menggunakan Kebijakan Kunci di AWS KMS dalam Panduan Developer AWS Key Management Service .

Untuk memberikan izin AWSServiceRoleForEC2SpotFleet peran untuk menggunakan CMK

- Gunakan perintah create-grant untuk menambahkan hibah ke CMK dan untuk menentukan prinsipal (peran terkait layanan Ar AWSServiceRoleForEC2Spot mada) yang diberi izin untuk melakukan operasi yang diizinkan oleh hibah. CMK ditentukan oleh parameter `key-id` dan ARN CMK. Prinsip ditentukan oleh `grantee-principal` parameter dan ARN dari peran terkait AWSServiceRoleForEC2Spot layanan Armada.

  ```
  aws kms create-grant \
      --region us-east-1 \
      --key-id arn:aws:kms:us-
  east-1:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab \
      --grantee-principal arn:aws:iam::111122223333:role/aws-service-role/
  spotfleet.amazonaws.com/AWSServiceRoleForEC2SpotFleet \
      --operations "Decrypt" "Encrypt" "GenerateDataKey"
   "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
   "ReEncryptTo"
  ```

## Service-linked peran untuk Instans Spot

Amazon EC2 menggunakan peran terkait layanan bernamaAWSServiceRoleForEC2Spot untuk meluncurkan dan mengelola Instans Spot atas nama Anda. Untuk informasi selengkapnya, lihat Service-linked peran untuk permintaan Instance Spot.

## Peran IAM untuk menandai Armada Spot

Peran IAM `aws-ec2-spot-fleet-tagging-role` memberikan izin ke Armada Spot untuk menandai permintaan, instans, dan volume Armada Spot. Untuk informasi selengkapnya, lihat Menandai permintaan Spot Fleet baru atau yang sudah ada serta instans serta volume yang diluncurkan.

> **Important**
>
> Jika Anda memilih untuk menandai instans di armada dan Anda juga memilih untuk mempertahankan kapasitas target (permintaan Armada Spot bertipe `maintain`), perbedaan izin yang ditetapkan untuk pengguna dan `IamFleetRole` dapat menyebabkan perilaku penandaan instans yang tidak konsisten di armada. Jika `IamFleetRole` tidak menyertakan izin `CreateTags`, beberapa instans yang diluncurkan oleh armada mungkin tidak akan ditandai. Sementara kami berusaha memperbaiki inkonsistensi ini, untuk memastikan bahwa semua instans yang diluncurkan oleh armada telah ditandai, kami menyarankan Anda menggunakan peran `aws-ec2-spot-fleet-tagging-role` untuk `IamFleetRole`. Atau, untuk menggunakan peran yang ada, lampirkan Kebijakan `AmazonEC2SpotFleetTaggingRole` AWS Terkelola ke peran yang ada. Jika tidak, Anda perlu menambahkan izin `CreateTags` secara manual untuk kebijakan yang ada.

Guna membuat peran IAM untuk menandai Armada Spot

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Pilih Buat peran.
4. Pada halaman Pilih entitas tepercaya, di bawah Tipe entitas tepercaya, pilih Layanan AWS .
5. Di bawah Kasus penggunaan, dari Kasus penggunaan untuk layanan AWS lain, pilih EC2, lalu pilih EC2 - Penandaan Armada Spot.
6. Pilih Berikutnya.

7. Pada halaman Tambahkan izin, pilih Berikutnya.
8. Pada Nama, tinjau, dan buat, untuk Nama peran, masukkan nama untuk peran (misalnya, **`aws-ec2-spot-fleet-tagging-role`**).
9. Tinjau informasi di halaman tersebut, lalu pilih Buat peran.

Cross-service pencegahan wakil yang bingung

Masalah confused deputy adalah masalah keamanan saat entitas yang tidak memiliki izin untuk melakukan suatu tindakan dapat memaksa entitas yang lebih berhak untuk melakukan tindakan tersebut. Sebaiknya gunakan kunci konteks kondisi global `aws:SourceArn` dan `aws:SourceAccount` dalam kebijakan kepercayaan `aws-ec2-spot-fleet-tagging-role` untuk membatasi izin yang diberikan Armada Spot pada layanan lain ke sumber daya.

Untuk menambahkan kunci SourceAccount kondisi aws: SourceArn dan aws: ke kebijakan kepercayaan **`aws-ec`** 2-spot-fleet-tagging-role

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran.
3. Temukan `aws-ec2-spot-fleet-tagging-role` yang Anda buat sebelumnya dan pilih tautan (bukan kotak centang).
4. Di bawah Ringkasan, pilih tab Hubungan kepercayaan, lalu pilih Edit kebijakan kepercayaan.
5. Dalam pernyataan JSON, tambahkan elemen `Condition` yang berisi kunci konteks kondisi global `aws:SourceAccount` dan `aws:SourceArn` untuk mencegah masalah confused deputy, sebagai berikut:

```
"Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:ec2:us-east-1:111122223333:spot-fleet-request/
sfr-*"
      },
      "StringEquals": {
        "aws:SourceAccount": "111122223333"
      }
```

> **Note**
>
> Jika nilai `aws:SourceArn` berisi ID akun Anda dan Anda menggunakan kedua kunci konteks kondisi global tersebut, nilai `aws:SourceAccount` dan akun di nilai `aws:SourceArn` harus menggunakan ID akun yang sama saat digunakan dalam pernyataan kebijakan yang sama.

Kebijakan kepercayaan terakhir adalah sebagai berikut:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "spotfleet.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:ec2:us-east-1:111122223333:spot-fleet-
request/sfr-*"
      },
      "StringEquals": {
        "aws:SourceAccount": "111122223333"
      }
    }
  }
}
```

6. Pilih Perbarui kebijakan.

Tabel berikut memberikan nilai potensial untuk `aws:SourceArn` guna membatasi ruang lingkup `aws-ec2-spot-fleet-tagging-role` Anda dalam berbagai tingkat kekhususan.

| Operasi API | Layanan yang dipanggil | Lingkup | **aws:SourceArn** |
|---|---|---|---|
| RequestSpotFleet | AWS STS (AssumeRole ) | Batasi kemampuan AssumeRole pada aws-ec2-spot-fleet-tagging-role untuk spot-fleet-requests di akun yang ditentukan. | arn:aws:ec2:*: *123456789012* :spot-fleet-request/sfr-* |
| RequestSpotFleet | AWS STS (AssumeRole ) | Batasi kemampuan AssumeRole pada aws-ec2-spot-fleet-tagging-role untuk spot-fleet-requests di akun yang ditentukan dan Wilayah yang ditentukan. Perhatikan bahwa peran ini tidak akan dapat digunakan di Wilayah lain. | arn:aws:ec2: *us-east-1* :*123456789012* :spot-fleet-request/sfr-* |
| RequestSpotFleet | AWS STS (AssumeRole ) | Batasi kemampuan AssumeRole di aws-ec2-spot-fleet-tagging-role hanya pada tindakan yang memengaruhi armada sfr-11111111-1111-1111-1111-111111111111. Perhatikan bahwa peran ini mungkin tidak dapat digunakan untuk | arn:aws:ec2: *us-east-1* :*123456789012* :spot-fleet-request/sfr-*11111111-1111-1111-1111-111111111111* |

| Operasi API | Layanan yang dipanggil | Lingkup | **aws:SourceArn** |
| --- | --- | --- | --- |
| | | Armada Spot lainnya. Selain itu, peran ini tidak dapat digunakan untuk meluncurkan Armada Spot baru melalui request-spot-fleet. | |

## Membuat Armada Spot

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Dengan menggunakan Konsol Manajemen AWS, Anda dapat dengan cepat membuat permintaan Spot Fleet dengan memilih hanya AMI dan total kapasitas target yang Anda inginkan. Amazon EC2 akan mengonfigurasi armada yang paling sesuai dengan kebutuhan Anda dan mengikuti praktik terbaik Spot. Atau, Anda dapat mengubah salah satu pengaturan default.

Jika Anda ingin menyertakan On-Demand Instans dalam armada Anda, Anda harus menentukan template peluncuran dalam permintaan Anda dan menentukan On-Demand kapasitas yang Anda inginkan.

Armada meluncurkan On-Demand Instans saat kapasitas tersedia, dan meluncurkan Instans Spot saat harga maksimum melebihi harga Spot dan kapasitas yang tersedia.

Jika armada Anda menyertakan Instans Spot dan jenisnya`maintain`, Amazon EC2 akan mencoba mempertahankan kapasitas target armada Anda saat Instans Spot terganggu.

Izin yang diperlukan

Untuk informasi selengkapnya, lihat the section called “Izin Armada Spot”.

Tugas

- Buat permintaan Spot Fleet dengan cepat
- Membuat permintaan Spot Fleet menggunakan parameter yang ditentukan
- Membuat Armada Spot yang menggantikan Instans Spot yang tidak sehat

## Buat permintaan Spot Fleet dengan cepat

Ikuti langkah-langkah berikut untuk membuat permintaan Spot Fleet dengan cepat menggunakan konsol Amazon EC2.

Untuk membuat permintaan Spot Fleet menggunakan pengaturan yang disarankan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Jika Anda baru mengenal Spot, Anda akan melihat halaman sambutan; pilih Mulai. Jika tidak, pilih Buat Permintaan Armada Spot.
4. Di bawah Parameter peluncuran, pilih Konfigurasikan parameter peluncuran secara manual.
5. Untuk AMI, pilih AMI.
6. Di bawah Kapasitas target, untuk Total kapasitas target, tentukan jumlah unit yang akan diminta. Untuk jenis unit, Anda dapat memilih Instans, vCPU, atau Memori (GiB).
7. Di bawah Sekilas permintaan armada Anda, tinjau konfigurasi armada Anda, dan pilih Luncurkan.

## Membuat permintaan Spot Fleet menggunakan parameter yang ditentukan

Anda dapat membuat Armada Spot menggunakan parameter yang Anda tentukan.

Console

Untuk membuat permintaan Spot Fleet menggunakan parameter yang ditentukan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Jika Anda baru mengenal Spot, Anda akan melihat halaman sambutan; pilih Mulai. Jika tidak, pilih Buat Permintaan Armada Spot.

4. Untuk parameter Launch, Anda dapat mengkonfigurasi parameter peluncuran secara manual atau Anda dapat menggunakan template peluncuran, sebagai berikut:

   a. [Konfigurasi secara manual] Untuk menentukan parameter peluncuran di konsol Amazon EC2, pilih Konfigurasikan parameter peluncuran secara manual, lalu lakukan hal berikut:

      i. Untuk AMI, pilih salah satu AMI dasar yang disediakan oleh AWS, atau pilih Cari AMI untuk menggunakan AMI dari komunitas pengguna kami AWS Marketplace, atau salah satu dari Anda sendiri.

      > **Note**
      >
      > Jika AMI yang ditentukan dalam parameter peluncuran dideregistrasi atau dinonaktifkan, tidak ada instans baru yang dapat diluncurkan dari AMI. Untuk armada yang diatur untuk mempertahankan kapasitas target, kapasitas target tidak akan dipertahankan.

      ii. (Opsional) Untuk Nama pasangan kunci, pilih pasangan kunci yang ada atau buat yang baru.

      [Pasangan kunci yang ada] Pilih pasangan kunci.

      [Pasangan kunci baru] Pilih Buat pasangan kunci baru untuk membuka halaman Pasangan kunci. Setelah selesai, kembali ke halaman Permintaan Spot dan segarkan daftar.

      iii. (Opsional) Perluas Parameter peluncuran tambahan, dan lakukan hal berikut:

         A. (Opsional) Untuk mengaktifkan pengoptimalan Amazon EBS, untuk EBS-optimized, pilih Luncurkan EBS-optimized inst ans.

         B. (Opsional) Guna menambahkan penyimpanan tingkat blok sementara untuk instans Anda, untuk Penyimpanan instans, pilih Lampirkan saat peluncuran.

         C. (Opsional) Untuk menambahkan penyimpanan, pilih Tambahkan volume baru, dan tentukan volume penyimpanan instans tambahan atau volume Amazon EBS, tergantung pada tipe instans.

         D. (Opsional) Secara default, pemantauan dasar diaktifkan untuk instans Anda. Untuk mengaktifkan pemantauan terperinci, untuk Pem antauan, pilih Aktifkan pemantauan CloudWatch terperinci.

E. (Opsional) Guna menjalankan Instans Spot Khusus, untuk Penghunian, pilih Khusus - jalankan instans khusus.

F. (Opsional) Untuk Grup keamanan, pilih satu atau beberapa grup keamanan atau buat yang baru.

[Grup keamanan yang ada] Pilih satu atau beberapa grup keamanan.

[Grup keamanan baru] Pilih Buat grup keamanan baru untuk membuka halaman Grup Keamanan. Setelah selesai, kembali ke Permintaan Spot dan segarkan daftar.

G. (Opsional) Untuk membuat instans Anda dapat dijangkau dari internet, untuk IP Publik Auto-assign IPv4, pilih Aktifkan.

H. (Opsional) Guna meluncurkan Instans Spot Anda dengan peran IAM, untuk Profil instans IAM, pilih peran tersebut.

I. (Opsional) Untuk menjalankan skrip start-up, salin skrip tersebut ke Data pengguna.

J. (Opsional) Untuk menambahkan tanda, pilih Buat tanda dan masukkan kunci serta nilai untuk tanda tersebut, lalu pilih Buat. Ulangi hal itu untuk setiap tanda.

Untuk setiap tanda, guna menandai instans dan permintaan Armada Spot dengan tanda yang sama, pastikan bahwa Instans serta Armada telah dipilih. Untuk menandai instans yang diluncurkan oleh armada saja, hapus Armada. Untuk menandai permintaan Armada Spot saja, hapus Instans.

b. [Template luncurkan] Untuk menggunakan konfigurasi yang Anda buat dalam template peluncuran, pilih Gunakan template peluncuran, dan untuk Lun curkan template, pilih template peluncuran.

> **Note**
>
> Jika Anda menginginkan On-Demand kapasitas di Armada Spot Anda, Anda harus menentukan template peluncuran.

5. Untuk detail permintaan tambahan, lakukan hal berikut:

a. Tinjau detail permintaan tambahan. Untuk membuat perubahan, hapus Terapkan default.

b. (Opsional) Untuk Peran armada IAM, Anda dapat menggunakan peran default atau memilih peran yang berbeda. Untuk menggunakan peran default setelah mengubah peran, pilih Gunakan peran default.

c. (Opsional) Untuk membuat permintaan yang hanya berlaku selama jangka waktu tertentu, edit Permintaan berlaku mulai dan Permintaan berlaku sampai.

d. (Opsional) Secara default, Amazon EC2 mengakhiri Instans Spot Anda saat permintaan Spot Fleet kedaluwarsa. Agar Instans Spot tetap berjalan setelah permintaan Anda berakhir, hapus Akhiri instans saat permintaan kedaluwarsa.

e. (Opsional) Untuk mendaftarkan Instans Spot Anda dengan penyeimbang beban, pilih Terima lalu lintas dari satu atau beberapa penyeimbang beban dan pilih satu atau beberapa Penyeimbang Beban Klasik atau grup target.

6. Untuk Kapasitas target, lakukan hal berikut:

a. Di bawah Total kapasitas target, tentukan jumlah unit yang akan diminta. Untuk tipe unit, Anda dapat memilih Instans, vCPU, atau Memori (MiB). Untuk menentukan kapasitas target 0 sehingga Anda dapat menambahkan kapasitas nanti, Anda harus terlebih dahulu memilih Per  tahankan kapasitas target.

b. (Opsional) Untuk  Sertakan kapasitas On-Demand dasar, tentukan jumlah On-Demand unit yang akan diminta. Jumlahnya harus kurang dari Total kapasitas target. Amazon EC2 menghitung selisihnya, dan mengalokasikan selisih tersebut ke unit Spot yang akan diminta.

> **Important**
>
> Untuk menentukan On-Demand kapasitas opsional, Anda harus terlebih dahulu memilih template peluncuran.

c. (Opsional) Secara default, Amazon EC2 mengakhiri Instans Spot saat terganggu. Untuk mempertahankan kapasitas target, pilih Pertahankan kapasitas target. Anda kemudian dapat menentukan bahwa Amazon EC2 mengakhiri, menghentikan, atau hibernasi Instans Spot saat terganggu. Untuk melakukannya, pilih opsi yang sesuai dari Perilaku interupsi.

> **Note**
>
> Jika AMI yang ditentukan dalam parameter peluncuran dideregistrasi atau dinonaktifkan, tidak ada instans baru yang dapat diluncurkan dari AMI. Dalam hal ini, untuk armada yang ditetapkan untuk mempertahankan kapasitas target, kapasitas target tidak akan dipertahankan.

d. (Opsional) Untuk mengizinkan Armada Spot meluncurkan Instans Spot pengganti saat notifikasi penyeimbangan ulang instans dikeluarkan untuk Instans Spot yang ada di armada, pilih Penyeimbangan ulang kapasitas, lalu pilih strategi penggantian instans. Jika Anda memilih  Luncurkan sebelum mengakhiri, tentukan penundaan (dalam detik) sebelum Amazon EC2 mengakhiri instans lama. Untuk informasi selengkapnya, lihat Gunakan Penyeimbangan Kembali Kapasitas di Armada EC2 dan Spot Fleet untuk menggantikan Instans Spot yang berisiko.

e. (Opsional) Untuk mengontrol jumlah yang Anda bayarkan per jam untuk semua Instans Spot di armada, pilih Atur biaya maksimum untuk Instans Spot, lalu masukkan jumlah total maksimum yang ingin Anda bayarkan per jam. Jika jumlah total maksimum tercapai, Armada Spot akan berhenti meluncurkan Instans Spot meskipun belum memenuhi kapasitas target. Untuk informasi selengkapnya, lihat Tetapkan batas pengeluaran untuk Armada EC2 atau Spot Fleet Anda.

7. Untuk Jaringan, lakukan hal berikut:

   a. Untuk Jaringan, pilih VPC yang ada atau buat yang baru.

      [VPC yang Ada] Pilih VPC.

      [VPC Baru] Pilih Buat VPC baru untuk membuka konsol Amazon VPC. Setelah selesai, kembali ke layar ini dan segarkan daftar.

   b. (Opsional) Untuk Zona Keter  sediaan, biarkan Amazon EC2 memilih Zona Ketersediaan untuk Instans Spot Anda, atau tentukan satu atau beberapa Zona Ketersediaan.

      Jika Anda memiliki lebih dari satu subnet di Zona Ketersediaan, pilih subnet yang sesuai dari Subnet. Untuk menambahkan subnet, pilih Buat subnet baru untuk membuka konsol Amazon VPC. Setelah selesai, kembali ke layar ini dan segarkan daftar.

8. Untuk Persyaratan tipe instans, Anda dapat menentukan atribut instans dan membiarkan Amazon EC2 mengidentifikasi tipe instans optimal dengan atribut ini, atau Anda dapat

menentukan daftar instans. Untuk informasi selengkapnya, lihat Tentukan atribut untuk pemilihan tipe instans untuk EC2 Fleet atau Spot Fleet.

a. Jika Anda memilih Tentukan atribut instans yang cocok dengan persyaratan komputasi Anda, tentukan atribut instans sebagai berikut:

   i. Untuk vCPU, masukkan jumlah minimum dan maksimum vCPU yang diinginkan. Untuk menentukan tidak ada batas, pilih Tidak ada minimum atau Tanpa maksimum, atau keduanya.

   ii. Untuk Memori (GiB), masukkan jumlah memori minimum dan maksimum yang diinginkan. Untuk menentukan tidak ada batas, pilih Tidak ada minimum atau Tanpa maksimum, atau keduanya.

   iii. (Opsional) Untuk atribut instance tambahan, Anda dapat secara opsional menentukan satu atau lebih atribut untuk mengungkapkan persyaratan komputasi Anda secara lebih rinci. Setiap atribut tambahan menambahkan batasan lebih lanjut ke permintaan Anda. Anda dapat menghilangkan atribut tambahan; ketika dihilangkan, nilai default digunakan. Untuk deskripsi setiap atribut dan nilai defaultnya, lihat get-spot-placement-scores.

   iv. (Opsional) Untuk menampilkan tipe instans dengan atribut tertentu, perluas Pratinjau tipe instans yang cocok. Untuk mengecualikan tipe instans agar tidak digunakan dalam permintaan Anda, pilih instans, lalu pilih Kecualikan tipe instans yang dipilih.

b. Jika Anda memilih Pilih tipe instans secara manual, Armada Spot menyediakan daftar default tipe instans. Untuk memilih tipe instans lainnya, pilih Tambahkan tipe instans, pilih tipe instans yang akan digunakan dalam permintaan Anda, dan pilih Pilih. Untuk menghapus tipe instans, pilih tipe instans dan pilih Hapus.

9. Untuk strategi Alokasi, pilih strategi alokasi Spot dan strategi On-Demand alokasi yang sesuai dengan kebutuhan Anda. Untuk informasi selengkapnya, lihat Gunakan strategi alokasi untuk menentukan bagaimana EC2 Fleet atau Spot Fleet memenuhi Spot dan kapasitas On-Demand.
10. Untuk Sekilas permintaan armada Anda, tinjau konfigurasi armada dan lakukan penyesuaian apa pun jika perlu.
11. (Opsional) Untuk mengunduh salinan konfigurasi peluncuran untuk digunakan dengan AWS CLI, pilih konfigurasi JSON.
12. Saat Anda siap meluncurkan Armada Spot Anda, pilih Luncurkan.

Tipe permintaan Armada Spot adalah `fleet`. Saat permintaan terpenuhi, permintaan tipe `instance` ditambahkan, di mana keadaannya `active` dan statusnya adalah `fulfilled`.

AWS CLI

Untuk membuat permintaan Spot Fleet

Gunakan perintah request-spot-fleet.

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://config.json
```

Untuk file konfigurasi contoh, lihat Contoh konfigurasi CLI Spot Fleet.

PowerShell

Untuk membuat permintaan Spot Fleet

Gunakan Request-EC2SpotFleet cmdlet. Contoh berikut meluncurkan Instans Spot dalam armada yang dioptimalkan kapasitas.

```
Request-EC2SpotFleet `
    -SpotFleetRequestConfig_TargetCapacity 50 `
    -SpotFleetRequestConfig_AllocationStrategy "CapacityOptimized" `
    -SpotFleetRequestConfig_IamFleetRole "arn:aws:iam::123456789012:role/my-spot-
fleet-role" `
    -SpotFleetRequestConfig_LaunchTemplateConfig @($launchConfig)
```

Tentukan konfigurasi peluncuran sebagai berikut, atur template peluncuran dan ganti properti yang Anda butuhkan. Untuk contoh konfigurasi, lihat Contoh konfigurasi CLI Spot Fleet.

```
$lcSpec = Amazon.EC2.Model.FleetLaunchTemplateSpecification
# To do - Set FleetLaunchTemplateSpecification properties
$lcOverrides = New-Object Amazon.EC2.Model.LaunchTemplateOverrides
# To do - Set LaunchTemplateOverrides properties
$launchConfig = New-Object Amazon.EC2.Model.LaunchTemplateConfig
$launchConfig.LaunchTemplateSpecification $lcSpec
$launchConfig.Overrides @($lcOverrides)
```

## Membuat Armada Spot yang menggantikan Instans Spot yang tidak sehat

Armada Spot memeriksa status kondisi Instans Spot di armada setiap dua menit. Status kondisi instans adalah `healthy` atau `unhealthy`.

Armada Spot menentukan status kondisi instans dengan menggunakan pemeriksaan status yang disediakan oleh Amazon EC2. Sebuah instans ditentukan sebagai `unhealthy` jika status pemeriksaan status instans atau pemeriksaan status sistemnya `impaired` dalam tiga kali pemeriksaan kondisi secara berturut–turut. Untuk informasi selengkapnya, lihat Pemeriksaan status untuk instans Amazon EC2.

Anda dapat mengonfigurasi armada untuk mengganti Instans Spot yang tidak sehat. Setelah mengaktifkan penggantian pemeriksaan kondisi, Instans Spot akan diganti jika dilaporkan sebagai `unhealthy`. Armada tersebut dapat berada di bawah kapasitas targetnya hingga beberapa menit saat Instans Spot yang tidak sehat sedang diganti.

Persyaratan

- Penggantian pemeriksaan kondisi hanya didukung untuk Armada Spot yang mempertahankan kapasitas target (armada tipe `maintain`), bukan untuk Armada Spot satu kali (armada tipe `request`).
- Penggantian pemeriksaan kondisi hanya didukung untuk Instans Spot. Fitur ini tidak didukung untuk On-Demand Instans.
- Anda dapat mengonfigurasi Armada Spot Fleet untuk mengganti instans yang tidak sehat hanya saat Anda membuatnya.
- Pengguna dapat menggunakan penggantian pemeriksaan kondisi hanya jika memiliki izin untuk memanggil tindakan `ec2:DescribeInstanceStatus`.

Console

Cara mengonfigurasi Armada Spot untuk mengganti Instans Spot yang tidak sehat

1. Ikuti langkah-langkah untuk membuat Armada Spot diMembuat permintaan Spot Fleet menggunakan parameter yang ditentukan.
2. Untuk mengonfigurasi armada untuk menggantikan Instans Spot yang tidak sehat, perluas Param eter peluncuran tambahan, dan di bawah Pemer iksaan kesehatan, pilih G anti instans yang tidak sehat. Untuk mengaktifkan opsi ini, Anda harus memilih Pertahankan kapasitas target terlebih dahulu.

AWS CLI

Cara mengonfigurasi Armada Spot untuk mengganti Instans Spot yang tidak sehat

Gunakan perintah request-spot-fleet dengan properti of. `ReplaceUnhealthyInstances` `SpotFleetRequestConfig`

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "lowestPrice",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
        "TargetCapacity": 10,
        "ReplaceUnhealthyInstances": true
    }
}
```

PowerShell

Untuk mengonfigurasi permintaan Spot Fleet untuk mengganti Instans Spot yang tidak sehat

Gunakan Request-EC2SpotFleet cmdlet dengan parameter. `-SpotFleetRequestConfig_ReplaceUnhealthyInstance`

```
-SpotFleetRequestConfig_ReplaceUnhealthyInstance $true
```

## Menandai permintaan Spot Fleet baru atau yang sudah ada serta instans serta volume yang diluncurkan

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Untuk membantu mengkategorikan dan mengelola permintaan Spot Fleet serta instans serta volume yang diluncurkan, Anda dapat menandainya dengan metadata khusus. Anda dapat menetapkan

tanda untuk permintaan Armada Spot saat Anda membuatnya, atau setelahnya. Demikian pula, Anda dapat menetapkan tag ke instance dan volume saat diluncurkan oleh armada, atau sesudahnya.

Saat Anda menandai permintaan armada, instans dan volume yang diluncurkan oleh armada tidak ditandai secara otomatis. Anda perlu menandai instans dan volume yang diluncurkan oleh armada secara eksplisit. Anda dapat memilih untuk menetapkan tag hanya untuk permintaan armada, atau hanya untuk instance yang diluncurkan oleh armada, atau hanya untuk volume yang dilampirkan ke instance yang diluncurkan oleh armada, atau ke semuanya.

> **Note**
>
> Anda hanya dapat menandai volume yang dilampirkan ke On-Demand Instans. Anda tidak dapat menandai volume yang dilampirkan ke Instans Spot.

Anda dapat menetapkan tanda menggunakan konsol Amazon EC2 atau alat baris perintah.

Untuk informasi selengkapnya tentang cara kerja tag, lihat Tandai sumber daya Amazon EC2 Anda.

Daftar Isi


- Prasyarat
- Menandai Armada Spot baru dan instans serta volume yang diluncurkannya
- Menandai Armada Spot yang ada
- Menampilkan tanda permintaan Armada Spot


## Prasyarat

Berikan izin kepada pengguna untuk menandai sumber daya. Untuk informasi selengkapnya, lihat Contoh: Memberi tanda pada sumber daya.

Berikan izin kepada pengguna untuk menandai sumber daya

Buat kebijakan IAM yang mencakup berikut hal berikut:

- Tindakan `ec2:CreateTags`. Tindakan ini memberikan izin kepada pengguna untuk membuat tanda.
- Tindakan `ec2:RequestSpotFleet`. Tindakan ini memberikan izin kepada pengguna untuk membuat permintaan Armada Spot.

- Untuk `Resource`, Anda harus menentukan `"*"`. Tindakan ini memungkinkan pengguna untuk menandai semua tipe sumber daya.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "TagSpotFleetRequest",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags",
                "ec2:RequestSpotFleet"
            ],
            "Resource": "*"
        }
    ]
}
```

**Important**

Resource-level izin saat ini tidak didukung untuk sumber `spot-fleet-request` daya. Jika Anda menentukan `spot-fleet-request` sebagai sumber daya, Anda akan mendapatkan pengecualian yang tidak sah saat mencoba menandai armada. Contoh berikut menggambarkan cara untuk tidak mengatur kebijakan.

```
{
    "Effect": "Allow",
    "Action": [
        "ec2:CreateTags",
        "ec2:RequestSpotFleet"
    ],
    "Resource": "arn:aws:ec2:us-east-1:111122223333:spot-fleet-request/*"
}
```

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.

- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

## Menandai Armada Spot baru dan instans serta volume yang diluncurkannya

Untuk menandai permintaan Spot Fleet baru serta instance serta volume yang diluncurkan menggunakan konsol

1. Ikuti prosedur Membuat permintaan Spot Fleet menggunakan parameter yang ditentukan.
2. Cara Anda menambahkan tag tergantung pada apakah Anda mengonfigurasi armada secara manual atau menggunakan template peluncuran.

   - Jika Anda mengonfigurasi armada secara manual, lakukan hal berikut:

     Untuk menambahkan tag, perluas Parameter peluncuran tambahan , pilih Buat tag, dan masukkan kunci dan nilai untuk tag. Ulangi hal itu untuk setiap tanda.

     Untuk setiap tanda, Anda dapat menandai permintaan Armada Spot dan instans dengan tanda yang sama. Untuk menandai keduanya, pastikan bahwa Instans dan Armada dipilih. Untuk menandai permintaan Armada Spot saja, hapus Instans. Untuk menandai instans yang diluncurkan oleh armada saja, hapus Armada.

     > **Note**
     >
     > Saat Anda mengonfigurasi armada secara manual, tidak ada opsi untuk menandai volume. Tag volume hanya didukung untuk volume yang dilampirkan ke On-Demand

Instans. Saat mengonfigurasi armada secara manual, Anda tidak dapat menentukan On-Demand Instans.

- Jika Anda menggunakan template peluncuran, lakukan hal berikut:

  Untuk menambahkan tag ke permintaan armada, di bawah  Tag, pilih  Buat Tag, dan masukkan kunci dan nilai untuk tag. Ulangi hal itu untuk setiap tanda.

  Untuk menandai sumber daya di armada Anda, Anda harus menentukan tag di template pel uncuran.

Untuk menandai permintaan Spot Fleet baru serta instance serta volume yang diluncurkan menggunakan AWS CLI

Untuk menandai permintaan Armada Spot saat Anda membuatnya, dan untuk menandai instans serta volume ketika diluncurkan oleh armada, konfigurasikan konfigurasi permintaan Armada Spot sebagai berikut:

Tanda permintaan Armada Spot:

- Tentukan tanda untuk permintaan Armada Spot di `SpotFleetRequestConfig`.
- Untuk `ResourceType`, tentukan `spot-fleet-request`. Jika Anda menentukan nilai lain, permintaan armada akan gagal.
- Untuk `Tags`, tentukan pasangan nilai-kunci. Anda dapat menentukan lebih dari satu pasangan nilai-kunci.

Tanda instans:

- Tentukan tanda untuk instans di `LaunchSpecifications`.
- Untuk `ResourceType`, tentukan `instance`. Jika Anda menentukan nilai lain, permintaan armada akan gagal.
- Untuk `Tags`, tentukan pasangan nilai-kunci. Anda dapat menentukan lebih dari satu pasangan nilai-kunci.

  Atau, Anda dapat menentukan tanda untuk instans di templat peluncuran yang direferensikan dalam permintaan Armada Spot.

Tanda volume:

- Tentukan tanda untuk volume di templat peluncuran yang direferensikan dalam permintaan Armada Spot. Penandaan volume `LaunchSpecifications` tidak didukung.

Dalam contoh berikut, permintaan Spot Fleet ditandai dengan dua tag: Kunci = Lingkungan dan Nilai = Produksi, dan Kunci = dan Nilai = 123. Cost-Center Instans yang diluncurkan oleh armada ditandai dengan satu tag (yang sama dengan salah satu tag untuk permintaan Spot Fleet): Key= Cost-Center dan Value=123.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchSpecifications": [
            {
                "ImageId": "ami-0123456789EXAMPLE",
                "InstanceType": "c4.large",
                "TagSpecifications": [
                    {
                        "ResourceType": "instance",
                        "Tags": [
                            {
                                "Key": "Cost-Center",
                                "Value": "123"
                            }
                        ]
                    }
                ]
            }
        ],
        "SpotPrice": "5",
        "TargetCapacity": 2,
        "TerminateInstancesWithExpiration": true,
        "Type": "maintain",
        "ReplaceUnhealthyInstances": true,
        "InstanceInterruptionBehavior": "terminate",
        "InstancePoolsToUseCount": 1,
        "TagSpecifications": [
            {
```

```
                "ResourceType": "spot-fleet-request",
                "Tags": [
                    {
                        "Key": "Environment",
                        "Value":"Production"
                    },
                    {
                        "Key": "Cost-Center",
                        "Value":"123"
                    }
                ]
            }
        ]
    }
}
```

Untuk menandai instans yang diluncurkan oleh Armada Spot menggunakan AWS CLI

Untuk menandai instans ketika diluncurkan oleh armada, Anda dapat menentukan tanda di templat peluncuran yang direferensikan dalam permintaan Armada Spot, atau Anda dapat menentukan tanda dalam konfigurasi permintaan Armada Spot sebagai berikut:

- Tentukan tanda untuk instans di `LaunchSpecifications`.
- Untuk `ResourceType`, tentukan `instance`. Jika Anda menentukan nilai lain, permintaan armada akan gagal.
- Untuk `Tags`, tentukan pasangan nilai-kunci. Anda dapat menentukan lebih dari satu pasangan nilai-kunci.

Dalam contoh berikut, instance yang diluncurkan oleh armada ditandai dengan satu tag: Key= Cost-Center dan Value=123.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "priceCapacityOptimized",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchSpecifications": [
            {
                "ImageId": "ami-0123456789EXAMPLE",
                "InstanceType": "c4.large",
```

```
                "TagSpecifications": [
                    {
                        "ResourceType": "instance",
                        "Tags": [
                            {
                                "Key": "Cost-Center",
                                "Value": "123"
                            }
                        ]
                    }
                ]
            }
        ],
        "SpotPrice": "5",
        "TargetCapacity": 2,
        "TerminateInstancesWithExpiration": true,
        "Type": "maintain",
        "ReplaceUnhealthyInstances": true,
        "InstanceInterruptionBehavior": "terminate",
        "InstancePoolsToUseCount": 1
    }
}
```

Untuk menandai volume yang dilampirkan ke On-Demand Instans yang diluncurkan oleh Armada Spot menggunakan AWS CLI

Untuk menandai volume saat dibuat oleh armada, Anda harus menentukan tanda di templat peluncuran yang direferensikan dalam permintaan Armada Spot.

> **Note**
>
> Tag volume hanya didukung untuk volume yang dilampirkan ke On-Demand Instans. Anda tidak dapat menandai volume yang dilampirkan ke Instans Spot.
> Penandaan volume `LaunchSpecifications` tidak didukung.

## Menandai Armada Spot yang ada

Untuk menandai permintaan Armada Spot yang sudah ada menggunakan konsol

Setelah membuat permintaan Spot Fleet, Anda dapat menambahkan tag ke permintaan armada menggunakan konsol.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih tab Tanda dan pilih Buat Tanda.

Untuk menandai permintaan Spot Fleet yang ada menggunakan AWS CLI

Anda dapat menggunakan perintah create-tags untuk menandai sumber daya yang ada. Dalam contoh berikut, permintaan Armada Spot yang ada ditandai dengan Kunci=tujuan dan Nilai=uji.

```
aws ec2 create-tags \
    --resources sfr-11112222-3333-4444-5555-66666EXAMPLE \
    --tags Key=purpose,Value=test
```

## Menampilkan tanda permintaan Armada Spot

Untuk menampilkan tanda permintaan Armada Spot menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda dan pilih tab Tanda.

Untuk menjelaskan tanda permintaan Armada Spot

Gunakan perintah describe-tags untuk melihat tanda sumber daya yang ditentukan. Dalam contoh berikut, Anda menjelaskan tanda untuk permintaan Armada Spot yang ditentukan.

```
aws ec2 describe-tags \
    --filters "Name=resource-id,Values=sfr-11112222-3333-4444-5555-66666EXAMPLE"
```

```
{
    "Tags": [
        {
            "Key": "Environment",
            "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
            "ResourceType": "spot-fleet-request",
            "Value": "Production"
```

```
        },
        {
            "Key": "Another key",
            "ResourceId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
            "ResourceType": "spot-fleet-request",
            "Value": "Another value"
        }
    ]
}
```

Anda juga dapat menampilkan tanda permintaan Armada Spot dengan menjelaskan permintaan Armada Spot.

Gunakan perintah describe-spot-fleet-requests untuk menampilkan konfigurasi permintaan Armada Spot yang ditentukan, yang menyertakan tanda apa pun yang ditentukan untuk permintaan armada.

```
aws ec2 describe-spot-fleet-requests \
    --spot-fleet-request-ids sfr-11112222-3333-4444-5555-66666EXAMPLE
```

```
{
    "SpotFleetRequestConfigs": [
        {
            "ActivityStatus": "fulfilled",
            "CreateTime": "2020-02-13T02:49:19.709Z",
            "SpotFleetRequestConfig": {
                "AllocationStrategy": "capacityOptimized",
                "OnDemandAllocationStrategy": "lowestPrice",
                "ExcessCapacityTerminationPolicy": "Default",
                "FulfilledCapacity": 2.0,
                "OnDemandFulfilledCapacity": 0.0,
                "IamFleetRole": "arn:aws:iam::111122223333:role/aws-ec2-spot-fleet-
tagging-role",
                "LaunchSpecifications": [
                    {
                        "ImageId": "ami-0123456789EXAMPLE",
                        "InstanceType": "c4.large"
                    }
                ],
                "TargetCapacity": 2,
                "OnDemandTargetCapacity": 0,
                "Type": "maintain",
                "ReplaceUnhealthyInstances": false,
```

```
                "InstanceInterruptionBehavior": "terminate"
            },
            "SpotFleetRequestId": "sfr-11112222-3333-4444-5555-66666EXAMPLE",
            "SpotFleetRequestState": "active",
            "Tags": [
                {
                    "Key": "Environment",
                    "Value": "Production"
                },
                {
                    "Key": "Another key",
                    "Value": "Another value"
                }
            ]
        }
    ]
}
```

## Menjelaskan permintaan Spot Fleet, instansnya, dan riwayat peristiwa

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Anda dapat menjelaskan konfigurasi Armada Spot, instans di Armada Spot, dan riwayat peristiwa Armada Spot Anda.

Console

Untuk menggambarkan Armada Spot Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda. ID dimulai dengan sfr-. Untuk melihat detail konfigurasi, pilih Deskripsi.
4. Guna membuat daftar Instans Spot untuk Armada Spot, pilih Instans.
5. Untuk menampilkan riwayat Armada Spot, pilih Riwayat.

AWS CLI

Untuk menjelaskan permintaan Spot Fleet Anda

Gunakan perintah describe-spot-fleet-requests.

```
aws ec2 describe-spot-fleet-requests \
    --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Untuk menjelaskan instans yang sedang berjalan untuk permintaan Spot Fleet yang ditentukan

Gunakan perintah describe-spot-fleet-instance.

```
aws ec2 describe-spot-fleet-instances \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE
```

Untuk menjelaskan riwayat acara untuk permintaan Spot Fleet yang ditentukan

Gunakan perintah describe-spot-fleet-request-history.

```
aws ec2 describe-spot-fleet-request-history \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --start-time 2024-05-18T00:00:00Z
```

PowerShell

Untuk menjelaskan permintaan Spot Fleet Anda

Gunakan Get-EC2SpotFleetRequest cmdlet.

```
Get-EC2SpotFleetRequest
```

Untuk menjelaskan instans yang sedang berjalan untuk permintaan Spot Fleet yang ditentukan

Gunakan Get-EC2SpotFleetInstance cmdlet.

```
Get-EC2SpotFleetInstance `
    -SpotFleetRequestId "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE"
```

Untuk menjelaskan riwayat acara untuk permintaan Spot Fleet yang ditentukan

Gunakan Get-EC2SpotFleetRequestHistory cmdlet.

```
Get-EC2SpotFleetRequestHistory `
    -SpotFleetRequestId "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -UtcStartTime 2024-05-18T00:00:00Z
```

## Memodifikasi permintaan Armada Spot

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Anda dapat memodifikasi permintaan Armada Spot yang aktif untuk menyelesaikan tugas berikut:

- Meningkatkan total kapasitas dan On-Demand porsi target
- Mengurangi total kapasitas dan On-Demand porsi target

Saat Anda meningkatkan kapasitas target total, Armada Spot meluncurkan Instans Spot tambahan sesuai dengan strategi al okasi untuk permintaan Spot Fleet. Saat Anda menambah On-Demand porsi, Armada Spot meluncurkan On-Demand Instans tambahan.

Saat Anda mengurangi total kapasitas target, Armada Spot membatalkan permintaan terbuka yang melebihi kapasitas target baru. Anda dapat meminta agar Armada Spot mengakhiri Instans Spot hingga ukuran armada mencapai kapasitas target yang baru. Jika strategi alokasinya adalah `diversified`, Armada Spot akan mengakhiri instans di seluruh kolam. Atau, Anda dapat meminta agar Armada Spot mempertahankan armada pada ukurannya saat ini, tetapi tidak mengganti Instans Spot apa pun yang terinterupsi atau yang Anda akhiri secara manual.

Pertimbangan-pertimbangan

- Anda tidak dapat memodifikasi permintaan Armada Spot satu kali. Anda hanya dapat memodifikasi permintaan Armada Spot jika memilih Pertahankan kapasitas target saat membuat permintaan Armada Spot.
- Ketika Armada Spot mengakhiri instans karena kapasitas target berkurang, instans tersebut akan menerima pemberitahuan interupsi Instans Spot.

Console

Untuk mengubah permintaan Spot Fleet

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih Tindakan, Modifikasi kapasitas target.
5. Dalam Modifikasi kapasitas target, lakukan hal berikut:
   a. Masukkan kapasitas dan On-Demand porsi target baru.
   b. (Opsional) Jika Anda menurunkan kapasitas target tetapi ingin mempertahankan armada pada ukurannya saat ini, hapus Akhiri instans.
   c. Pilih Kirim.

AWS CLI

Untuk mengubah permintaan Spot Fleet

Gunakan perintah modify-spot-fleet-request untuk memperbarui kapasitas target permintaan Armada Spot yang ditentukan.

```
aws ec2 modify-spot-fleet-request \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity 20
```

Anda dapat mengubah perintah sebelumnya sebagai berikut untuk mengurangi kapasitas target Armada Spot yang ditentukan tanpa mengakhiri Instans Spot sebagai akibatnya.

```
aws ec2 modify-spot-fleet-request \
    --spot-fleet-request-id sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --target-capacity 10 \
    --excess-capacity-termination-policy NoTermination
```

PowerShell

Untuk mengubah permintaan Spot Fleet

Gunakan Edit-EC2SpotFleetRequest cmdlet untuk memperbarui kapasitas target permintaan Spot Fleet yang ditentukan.

```
Edit-EC2SpotFleetRequest `
    -SpotFleetRequestId "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TargetCapacity 20
```

Anda dapat mengubah perintah sebelumnya sebagai berikut untuk mengurangi kapasitas target Armada Spot yang ditentukan tanpa mengakhiri Instans Spot sebagai akibatnya.

```
Edit-EC2SpotFleetRequest `
    -SpotFleetRequestId "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TargetCapacity 20 `
    -ExcessCapacityTerminationPolicy "NoTermination"
```

## Membatalkan (menghapus) permintaan Spot Fleet

**Important**

Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Jika Anda tidak lagi memerlukan Armada Spot, Anda dapat membatalkan permintaan Spot Fleet, yang menghapus permintaan tersebut. Setelah Anda membatalkan permintaan armada, semua permintaan Spot yang terkait dengan armada juga dibatalkan, sehingga tidak ada Instans Spot baru yang diluncurkan.

Saat membatalkan permintaan Armada Spot, Anda juga harus menentukan apakah ingin mengakhiri semua instans. Ini termasuk On-Demand Instans dan Instans Spot.

**Warning**

Mengakhiri instance bersifat permanen dan tidak dapat diubah.
Setelah Anda menghentikan instans, Anda tidak dapat lagi terhubung ke instans, dan instans tidak dapat dipulihkan. Semua volume Amazon EBS terlampir yang dikonfigurasi untuk dihapus pada penghentian juga dihapus secara permanen dan tidak dapat dipulihkan. Semua

data yang disimpan pada volume penyimpanan instans hilang secara permanen. Untuk informasi selengkapnya, lihat Cara kerja penghentian instans.
Sebelum Anda menghentikan instans, pastikan Anda telah mencadangkan semua data yang perlu Anda simpan setelah penghentian ke penyimpanan persisten.

Jika Anda menentukan bahwa instans harus diakhiri saat permintaan armada dibatalkan, permintaan armada akan memasuki status `cancelled_terminating`. Jika tidak, armada masuk ke status `cancelled_running` dan instans terus berjalan hingga diinterupsi atau Anda mengakhirinya secara manual.

Pembatasan

- Anda dapat membatalkan hingga 100 armada dalam satu permintaan. Jika Anda melebihi jumlah yang ditentukan, tidak ada armada yang dibatalkan.

Console

Untuk membatalkan (menghapus) permintaan Spot Fleet

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih Tindakan, Batalkan permintaan.
5. Di kotak dialog Batalkan permintaan Spot, lakukan hal berikut:
   a. Untuk menghentikan instans terkait pada saat yang sama dengan membatalkan permintaan Spot Fleet, biarkan kotak centang Hentikan instans terpilih. Untuk membatalkan permintaan Spot Fleet tanpa mengakhiri instans terkait, kosongkan kotak centang Hentikan instans.
   b. Pilih Konfirmasi.

AWS CLI

Untuk membatalkan (menghapus) permintaan Spot Fleet dan menghentikan instansnya

Gunakan perintah cancel-spot-fleet- requests dengan opsi. `--terminate-instances`

```
aws ec2 cancel-spot-fleet-requests \
    --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --terminate-instances
```

Berikut ini adalah output contoh.

```
{
    "SuccessfulFleetRequests": [
        {
            "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
            "CurrentSpotFleetRequestState": "cancelled_terminating",
            "PreviousSpotFleetRequestState": "active"
        }
    ],
    "UnsuccessfulFleetRequests": []
}
```

Untuk membatalkan (menghapus) permintaan Spot Fleet tanpa mengakhiri instansnya

Ubah contoh sebelumnya dengan menggunakan `--no-terminate-instances` opsi sebagai gantinya.

```
aws ec2 cancel-spot-fleet-requests \
    --spot-fleet-request-ids sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE \
    --no-terminate-instances
```

Berikut ini adalah output contoh.

```
{
    "SuccessfulFleetRequests": [
        {
            "SpotFleetRequestId": "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE",
            "CurrentSpotFleetRequestState": "cancelled_running",
            "PreviousSpotFleetRequestState": "active"
        }
    ],
    "UnsuccessfulFleetRequests": []
}
```

PowerShell

Untuk membatalkan (menghapus) permintaan Spot Fleet dan menghentikan instansnya

Gunakan Stop-EC2SpotFleetRequest cmdlet dengan parameter. `-TerminateInstance`

```
Stop-EC2SpotFleetRequest `
    -SpotFleetRequestId "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TerminateInstance $true
```

Untuk membatalkan (menghapus) permintaan Spot Fleet tanpa mengakhiri instansnya

Ubah contoh sebelumnya dengan mengubah nilai `-TerminateInstance` parameter.

```
Stop-EC2SpotFleetRequest `
    -SpotFleetRequestId "sfr-73fbd2ce-aa30-494c-8788-1cee4EXAMPLE" `
    -TerminateInstance $false
```

## Memahami penskalaan otomatis untuk Spot Fleet

> **Important**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Penskalaan otomatis memungkinkan Armada Spot Anda meningkatkan atau mengurangi kapasitas target berdasarkan permintaan. Dengan penskalaan otomatis, Spot Fleet dapat meluncurkan instans (menskalakan) atau menghentikan instans (menskalakan) dalam rentang tertentu, sebagai respons terhadap satu atau lebih kebijakan penskalaan.

Penskalaan otomatis untuk Spot Fleet dimungkinkan oleh kombinasi Amazon EC2, Amazon CloudWatch, dan Application Auto Scaling API. Permintaan Spot Fleet dibuat dengan Amazon EC2, alarm dibuat dengan CloudWatch, dan kebijakan penskalaan dibuat dengan Penskalaan Otomatis Aplikasi.

Jenis penskalaan otomatis

Armada Spot mendukung tipe penskalaan otomatis berikut:

- Penskalaan pelacakan target - Meningkatkan atau mengurangi

kapasitas armada saat ini dengan menargetkan nilai untuk metrik tertentu. Ini mirip dengan cara termostat Anda mempertahankan suhu rumah Anda — Anda memilih suhu yang diinginkan dan termostat melakukan sisanya.

- Penskalaan bertahap – Meningkatkan atau menurunkan kapasitas armada saat ini berdasarkan set penyesuaian penskalaan, yang disebut dengan penyesuaian langkah, yang bervariasi berdasarkan ukuran pelanggaran alarm.
- Penskalaan Terjadwal – Meningkatkan atau mengurangi kapasitas armada saat ini berdasarkan tanggal dan waktu.

## Pertimbangan-pertimbangan

Saat menggunakan penskalaan otomatis untuk Armada Spot Anda, pertimbangkan hal berikut:

- Pembobotan instans — Jika Anda menggunakan bobot instans, perlu diingat bahwa Spot Fleet dapat melebihi kapasitas target sesuai kebutuhan. Kapasitas yang terpenuhi dapat berupa angka titik mengambang, tetapi kapasitas target harus berupa bilangan bulat, sehingga Armada Spot membulatkan ke bilangan bulat berikutnya. Anda harus mempertimbangkan perilaku ini jika Anda melihat hasil dari kebijakan penskalaan saat alarm dipicu. Sebagai contoh, misalkan kapasitas target adalah 30, kapasitas yang terpenuhi adalah 30,1, dan kebijakan penskalaan dikurangi 1. Apabila alarm dipicu, proses penskalaan otomatis akan mengurangi 1 dari 30,1 untuk mendapatkan 29,1, kemudian membulatkannya menjadi 30, sehingga tidak ada tindakan penskalaan yang dilakukan. Sebagai contoh lain, misalkan Anda memilih bobot instans 2, 4, dan 8, serta kapasitas target 10, tetapi tidak ada instans bobot 2 yang tersedia sehingga Armada Spot akan menyediakan instans bobot 4 dan 8 untuk kapasitas terpenuhi sebesar 12. Jika kebijakan penskalaan mengurangi kapasitas target sebesar 20% dan alarm dipicu, proses penskalaan otomatis akan mengurangi 12*0,2 dari 12 untuk mendapatkan 9,6, kemudian membulatkannya menjadi 10, sehingga tidak ada tindakan penskalaan yang dilakukan.
- Periode cooldown — Kebijakan penskalaan yang Anda buat untuk Spot Fleet mendukung periode cooldown. Periode ini adalah jumlah detik setelah aktivitas penskalaan selesai saat aktivitas penskalaan terkait pemicu sebelumnya dapat memengaruhi peristiwa penskalaan di masa mendatang. Untuk kebijakan penskalaan ke luar, selama periode pendinginan berlaku, kapasitas yang telah ditambahkan oleh peristiwa penskalaan ke luar sebelumnya yang memulai pendinginan dihitung sebagai bagian dari kapasitas yang diinginkan untuk penskalaan ke luar berikutnya. Tujuannya adalah untuk terus (tetapi tidak berlebihan) menskalakan ke luar. Untuk kebijakan penskalaan ke dalam, periode pendinginan digunakan untuk memblokir permintaan penskalaan ke dalam berikutnya hingga kedaluwarsa. Tujuannya adalah untuk menskalakan ke

dalam secara konservatif guna melindungi ketersediaan aplikasi Anda. Namun, jika alarm lain memicu kebijakan penskalaan ke luar selama periode pendinginan setelah penskalaan ke dalam, penskalaan otomatis akan segera mengurangi target yang dapat diskalakan.

- Gunakan pemantauan terperinci — Sebaiknya Anda menskalakan berdasarkan metrik instans dengan frekuensi 1 menit karena hal itu memastikan respons yang lebih cepat terhadap perubahan pemanfaatan. Penskalaan pada metrik dengan frekuensi 5 menit dapat menyebabkan waktu respons yang lebih lambat dan penskalaan pada data metrik yang sudah usang. Untuk mengirim data metrik untuk instance Anda CloudWatch dalam periode 1 menit, Anda harus secara khusus mengaktifkan pemantauan mendetail. Untuk informasi selengkapnya, lihat Kelola pemantauan terperinci untuk instans EC2 Anda dan Membuat permintaan Spot Fleet menggunakan parameter yang ditentukan.
- AWS CLI- Jika Anda menggunakan AWS CLI untuk mengonfigurasi penskalaan untuk Spot Fleet, Anda akan menggunakan perintah application-auto scaling.

## Izin IAM diperlukan untuk penskalaan otomatis Armada Spot

Penskalaan otomatis untuk Spot Fleet dimungkinkan oleh kombinasi Amazon EC2, Amazon CloudWatch, dan Application Auto Scaling API. Permintaan Spot Fleet dibuat dengan Amazon EC2, alarm dibuat dengan CloudWatch, dan kebijakan penskalaan dibuat dengan Penskalaan Otomatis Aplikasi. Selain izin IAM yang diperlukan untuk menggunakan Spot Fleet dan Amazon EC2, pengguna yang mengakses pengaturan penskalaan armada harus memiliki izin yang sesuai untuk layanan yang mendukung penskalaan otomatis.

Untuk menggunakan penskalaan otomatis untuk Spot Fleet, pengguna harus memiliki izin untuk menggunakan tindakan yang ditunjukkan dalam contoh kebijakan berikut.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "application-autoscaling:*",
                "ec2:DescribeSpotFleetRequests",
                "ec2:ModifySpotFleetRequest",
                "cloudwatch:DeleteAlarms",
```

```
                "cloudwatch:DescribeAlarmHistory",
                "cloudwatch:DescribeAlarms",
                "cloudwatch:DescribeAlarmsForMetric",
                "cloudwatch:GetMetricStatistics",
                "cloudwatch:ListMetrics",
                "cloudwatch:PutMetricAlarm",
                "cloudwatch:DisableAlarmActions",
                "cloudwatch:EnableAlarmActions",
                "iam:CreateServiceLinkedRole",
                "sns:CreateTopic",
                "sns:Subscribe",
                "sns:Get*",
                "sns:List*"
            ],
            "Resource": "*"
        }
    ]
}
```

Anda juga dapat membuat kebijakan IAM Anda sendiri yang memungkinkan izin yang lebih mendetail untuk panggilan ke API Penskalaan Otomatis Aplikasi. Untuk informasi selengkapnya, lihat Manajemen Identitas dan Akses untuk Penskalaan Otomatis Aplikasi di Panduan Pengguna Penskalaan Otomatis Aplikasi.

Layanan Penskalaan Otomatis Aplikasi juga memerlukan izin untuk menjelaskan Armada Spot dan CloudWatch alarm Anda, serta izin untuk mengubah kapasitas target Armada Spot atas nama Anda. Jika Anda mengaktifkan penskalaan otomatis untuk Armada Spot, fitur ini akan menciptakan peran tertaut layanan bernama `AWSServiceRoleForApplicationAutoScaling_EC2SpotFleetRequest`. Peran tertaut layanan ini memberikan izin Penskalaan Otomatis Aplikasi untuk mendeskripsikan alarm bagi kebijakan Anda, memantau kapasitas armada saat ini, dan memodifikasi kapasitas armada. Peran Armada Spot terkelola asli untuk Penskalaan Otomatis Aplikasi adalah`aws-ec2-spot-fleet-autoscale-role`, tetapi tidak lagi diperlukan. Peran tertaut layanan adalah peran default untuk Penskalaan Otomatis Aplikasi. Untuk informasi selengkapnya, lihat Service-linked peran untuk Penskalaan Otomatis Aplikasi di Panduan Pengguna Penskalaan Otomatis Aplikasi.

## Penskalaan pelacakan target: Menskalakan Armada Spot dengan menargetkan nilai untuk metrik tertentu

Dengan penskalaan pelacakan target, Anda membuat kebijakan penskalaan pelacakan target dengan memilih metrik dan menetapkan nilai target. Spot Fleet kemudian membuat dan mengelola CloudWatch alarm yang memicu kebijakan penskalaan, dan menghitung penyesuaian penskalaan berdasarkan metrik dan nilai target yang dipilih. Kebijakan penskalaan menyesuaikan kapasitas dengan menambahkan atau menghapus instans sesuai kebutuhan untuk menjaga metrik tetap pada, atau mendekati, nilai target yang ditentukan. Kebijakan pelacakan target tidak hanya menjaga metrik tetap dekat dengan nilai target, tetapi juga menyesuaikan dengan fluktuasi metrik karena pola beban yang berfluktuasi dan meminimalkan fluktuasi kapasitas yang cepat.

Anda dapat membuat beberapa kebijakan penskalaan pelacakan target untuk Armada Spot, asalkan setiap kebijakan menggunakan metrik yang berbeda. Skala armada berdasarkan kebijakan yang menentukan kapasitas armada terbesar. Ini memungkinkan Anda untuk mencakup beberapa skenario untuk memastikan kapasitas yang cukup untuk beban kerja aplikasi Anda.

Untuk memastikan ketersediaan aplikasi, armada menskalakan ke luar secara proporsional dengan metrik secepat mungkin, tetapi menskalakan ke dalam secara lebih bertahap.

Ketika Armada Spot mengakhiri Instans Spot karena kapasitas target berkurang, instans menerima pemberitahuan gangguan Instans Spot.

> **Note**
>
> Jangan mengedit atau menghapus CloudWatch alarm yang dikelola Spot Fleet untuk kebijakan penskalaan pelacakan target. Armada Spot menghapus alarm secara otomatis saat Anda menghapus kebijakan penskalaan pelacakan target.

### Prasyarat

- Permintaan Armada Spot harus memiliki permintaan tipe `maintain`. Penskalaan otomatis tidak didukung untuk permintaan tipe `request`.
- KonfigurasikanIzin IAM diperlukan untuk penskalaan otomatis Armada Spot.
- TinjauPertimbangan-pertimbangan.

Untuk mengonfigurasi kebijakan pelacakan target

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih tab Auto Scaling di dekat bagian bawah layar. Jika Anda memilih tautan untuk Armada Spot Anda, tidak ada tab; sebagai gantinya, gulir ke bawah ke bagian Auto Scaling.
5. Jika penskalaan otomatis tidak dikonfigurasi, pilih Konfigurasikan.
6. Gunakan Skalakan kapasitas antara guna mengatur kapasitas minimum dan maksimum untuk armada Anda. Penskalaan otomatis tidak menskalakan armada Anda di bawah kapasitas minimum atau di atas kapasitas maksimum.
7. Untuk Nama kebijakan, masukkan nama untuk kebijakan tersebut.
8. Pilih Metrik target.
9. Masukkan Nilai target untuk metrik.
10. Untuk Periode pendinginan, tentukan nilai baru (dalam detik) atau simpan default.
11. (Opsional) Untuk menghilangkan pembuatan kebijakan penskalaan berdasarkan konfigurasi saat ini, pilih Nonaktifkan penskalaan. Anda dapat membuat kebijakan penskalaan ke dalam menggunakan konfigurasi yang berbeda.
12. Pilih Simpan.

Untuk mengonfigurasi kebijakan pelacakan target menggunakan AWS CLI

1. Daftarkan permintaan Armada Spot sebagai target yang dapat diskalakan menggunakan perintah register-scalable-target.
2. Buat kebijakan penskalaan menggunakan perintah put-scaling-policy.

## Penskalaan langkah: Menskalakan Armada Spot menggunakan kebijakan penskalaan langkah

Dengan kebijakan penskalaan langkah, Anda menentukan CloudWatch alarm untuk memicu proses penskalaan. Misalnya, jika Anda ingin melakukan penskalaan ke luar saat pemanfaatan CPU mencapai tingkat tertentu, buat alarm menggunakan metrik `CPUUtilization` yang disediakan oleh Amazon EC2.

Saat membuat kebijakan penskalaan bertahap, Anda harus menentukan salah satu dari tipe penyesuaian penskalaan berikut:

- Tambah – Meningkatkan kapasitas target armada dengan jumlah unit kapasitas tertentu atau persentase tertentu dari kapasitas saat ini.
- Hapus – Mengurangi kapasitas target armada dengan jumlah unit kapasitas tertentu atau persentase tertentu dari kapasitas saat ini.
- Atur ke – Mengatur kapasitas target armada ke jumlah unit kapasitas yang ditentukan.

Saat alarm dipicu, proses penskalaan otomatis akan menghitung kapasitas target baru menggunakan kapasitas yang terpenuhi dan kebijakan penskalaan, lalu memperbarui kapasitas target yang sesuai. Sebagai contoh, misalkan kapasitas target dan kapasitas yang terpenuhi adalah 10 serta kebijakan penskalaan menambahkan 1. Saat alarm dipicu, proses penskalaan otomatis akan menambahkan 1 hingga 10 untuk mendapatkan 11, jadi Armada Spot meluncurkan 1 instans.

Ketika Armada Spot mengakhiri Instans Spot karena kapasitas target berkurang, instans menerima pemberitahuan gangguan Instans Spot.

Prasyarat

- Permintaan Armada Spot harus memiliki permintaan tipe `maintain`. Penskalaan otomatis tidak didukung untuk permintaan tipe `request`.
- KonfigurasikanIzin IAM diperlukan untuk penskalaan otomatis Armada Spot.
- Pertimbangkan CloudWatch metrik mana yang penting untuk aplikasi Anda. Anda dapat membuat CloudWatch alarm berdasarkan metrik yang disediakan oleh AWS atau metrik kustom Anda sendiri.
- Untuk AWS metrik yang akan Anda gunakan dalam kebijakan penskalaan, aktifkan CloudWatch pengumpulan metrik jika layanan yang menyediakan metrik tidak mengaktifkannya secara default.
- TinjauPertimbangan-pertimbangan.

Untuk membuat CloudWatch alarm

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, perluas Alarm dan pilih Semua alarm.
3. Pilih Buat alarm.
4. Di halaman Tentukan metrik dan kondisi, pilih Pilih metrik.

5. Pilih EC2 Spot, lalu Metrik Per mintaan Ar mada, lalu pilih metrik (misalnya, TargetCapacity), lalu pilih Pilih metrik.

   Halaman Tentukan metrik dan kondisi ditampilkan, yang menunjukkan grafik dan informasi lain tentang metrik yang Anda pilih.

6. Untuk Peri ode, pilih periode evaluasi untuk alarm, misalnya, 1 menit. Saat Anda mengevaluasi alarm, tiap periode akan digabungkan menjadi satu titik data.

   > **Note**
   >
   > Periode yang lebih pendek menghasilkan alarm yang lebih sensitif.

7. Untuk Kondisi, tentukan alarm dengan menentukan kondisi ambang batas. Misalnya, Anda dapat menentukan ambang batas untuk memicu alarm setiap kali nilai metrik lebih besar dari atau sama dengan 80 persen.
8. Di Konfigurasi tambahan, untuk Titik data ke alarm, tentukan banyaknya titik data (periode evaluasi) yang harus berada dalam status ALARM untuk memicu alarm, misalnya, 1 periode evaluasi atau 2 dari 3 periode evaluasi. Hal tersebut membuat alarm yang masuk ke status ALARM jika terjadi pelanggaran sebanyak itu secara berturut-turut. Untuk informasi selengkapnya, lihat Meng evaluasi alarm di Panduan CloudWatch Pengguna Amazon.
9. Untuk Penanganan data hilang, pilih salah satu opsi (atau biarkan default Perlakukan data yang hilang sebagai hilang). Untuk informasi selengkapnya, lihat Mengonfigurasi cara CloudWatch alarm memperlakukan data yang hilang di Panduan CloudWatch Pengguna Amazon.
10. Pilih Berikutnya.
11. (Opsional) Agar menerima notifikasi peristiwa penskalaan, untuk Notifikasi, Anda dapat memilih atau membuat topik Amazon SNS yang ingin Anda gunakan untuk menerima notifikasi. Jika tidak, Anda dapat menghapus notifikasi sekarang dan menambahkannya nanti sesuai kebutuhan.
12. Pilih Berikutnya.
13. Di bawah Tambahkan nama dan deskripsi, masukkan nama dan deskripsi untuk alarm dan pilih Ber ikutnya.
14. Pilih Buat alarm.

Untuk mengonfigurasi kebijakan penskalaan langkah untuk Armada Spot Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih tab Auto Scaling di dekat bagian bawah layar. Jika Anda memilih tautan untuk Armada Spot Anda, tidak ada tab; sebagai gantinya, gulir ke bawah ke bagian Auto Scaling.
5. Jika penskalaan otomatis tidak dikonfigurasi, pilih Konfigurasikan.
6. Gunakan Skalakan kapasitas antara guna mengatur kapasitas minimum dan maksimum untuk armada Anda. Kebijakan penskalaan tidak menskalakan armada Anda di bawah atau di atas kapasitas maksimum.
7. Di bawah Kebijakan penskalaan, untuk Jenis kebijakan, pilih Kebijakan penskalaan langkah.
8. Awalnya, Kebijakan penskalaan berisi kebijakan penskalaan langkah yang bernama ScaleUp dan ScaleDown. Anda dapat melengkapi kebijakan ini, atau memilih Hapus kebijakan untuk menghapusnya. Anda juga dapat memilih Tambahkan kebijakan.
9. Untuk menentukan kebijakan, lakukan hal berikut:

    a. Untuk Nama kebijakan, masukkan nama untuk kebijakan tersebut.
    b. Untuk Pemicu kebijakan, pilih alarm yang ada, atau pilih Buat alarm untuk membuka CloudWatch konsol Amazon dan membuat alarm.
    c. Untuk Modifikasi kapasitas, tentukan jumlah yang akan diskalakan serta batas bawah dan atas dari penyesuaian langkah. Anda dapat menambahkan atau menghapus sejumlah instans tertentu atau persentase ukuran armada yang ada, atau mengatur armada ke ukuran yang tepat.

        Misalnya, untuk membuat kebijakan penskalaan langkah yang meningkatkan kapasitas armada sebesar 30 persen, pilih Tambah, masukkan 30 di bidang berikutnya, lalu pilih persen. Secara default, batas bawah untuk kebijakan penambahan adalah ambang batas alarm, sedangkan batas atas adalah positif (+) tak terbatas. Secara default, batas atas untuk kebijakan penghapusan adalah ambang batas alarm, sedangkan batas bawah adalah negatif (-) tak terbatas.
    d. (Opsional) untuk menambahkan langkah lain, pilih Tambahkan langkah.
    e. Untuk Periode pendinginan, tentukan nilai baru (dalam detik) atau simpan default.

10. Pilih Simpan.

Untuk mengonfigurasi kebijakan penskalaan langkah untuk Armada Spot menggunakan AWS CLI

1. Daftarkan permintaan Armada Spot sebagai target yang dapat diskalakan menggunakan perintah register-scalable-target.
2. Buat kebijakan penskalaan menggunakan perintah put-scaling-policy.
3. Buat alarm yang memicu kebijakan penskalaan menggunakan perintah put-metric-alarm.

## Penskalaan terjadwal: Skalakan Armada Spot sesuai jadwal

Menskalakan armada Anda sesuai jadwal memungkinkan Anda untuk menskalakan aplikasi Anda sebagai respons terhadap perubahan permintaan yang dapat diprediksi. Dengan membuat tindakan terjadwal, Anda dapat menginstruksikan Spot Fleet untuk melakukan aktivitas penskalaan pada waktu tertentu. Untuk membuat tindakan terjadwal, Anda harus menentukan Armada Spot yang ada, waktu ketika aktivitas penskalaan harus terjadi, dan kapasitas minimum dan maksimum yang diinginkan. Tindakan terjadwal dapat dikonfigurasi untuk skala sekali atau pada jadwal berulang. Jika perlu diubah, Anda dapat mengedit atau menghapus tindakan terjadwal.

Prasyarat

- Tindakan terjadwal hanya dapat dibuat untuk Armada Spot yang ada. Anda tidak dapat membuat tindakan terjadwal saat membuat Spot Fleet.
- Permintaan Armada Spot harus memiliki permintaan tipe `maintain`. Penskalaan otomatis tidak didukung untuk permintaan tipe `request`.
- KonfigurasikanIzin IAM diperlukan untuk penskalaan otomatis Armada Spot.
- TinjauPertimbangan-pertimbangan.

Untuk membuat tindakan terjadwal satu kali

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih tab Sch eduled Scaling di dekat bagian bawah layar. Jika Anda memilih tautan untuk Armada Spot Anda, tidak ada tab; sebagai gantinya, gulir ke bawah ke bagian Penskalaan Ter jadwal.
5. Pilih Buat tindakan terjadwal.

6. Untuk Nama, tentukan nama untuk tindakan terjadwal.
7. Masukkan nilai untuk Kapasitas minimum, Kapasitas maksimum, atau keduanya.
8. Untuk Perulangan, pilih Sekali.
9. (Opsional) Pilih tanggal dan waktu untuk Waktu mulai, Waktu berakhir, atau keduanya.
10. Pilih Buat.

Untuk membuat tindakan terjadwal berulang

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih tab Sch eduled Scaling di dekat bagian bawah layar. Jika Anda memilih tautan untuk Armada Spot Anda, tidak ada tab; sebagai gantinya, gulir ke bawah ke bagian Penskalaan Ter jadwal.
5. Untuk Nama, tentukan nama untuk tindakan terjadwal.
6. Masukkan nilai untuk Kapasitas minimum, Kapasitas maksimum, atau keduanya.
7. Untuk Perulangan, pilih salah satu jadwal yang telah ditentukan sebelumnya (misalnya, Setiap hari), atau pilih Kustom dan masukkan ekspresi cron. Untuk informasi selengkapnya tentang ekspresi cron yang didukung oleh penskalaan terjadwal, lihat ekspresi Cron di Panduan EventBridge Pengguna Amazon.
8. (Opsional) Pilih tanggal dan waktu untuk Waktu mulai, Waktu berakhir, atau keduanya.
9. Pilih Kirim.

Untuk mengedit tindakan terjadwal

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih tab Sch eduled Scaling di dekat bagian bawah layar. Jika Anda memilih tautan untuk Armada Spot Anda, tidak ada tab; sebagai gantinya, gulir ke bawah ke bagian Penskalaan Ter jadwal.
5. Pilih tindakan terjadwal dan pilih Tindakan, Edit.
6. Lakukan perubahan yang diperlukan dan pilih Kirim.

Untuk menghapus tindakan terjadwal

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Permintaan Spot.
3. Pilih permintaan Armada Spot Anda.
4. Pilih tab Sch eduled Scaling di dekat bagian bawah layar. Jika Anda memilih tautan untuk Armada Spot Anda, tidak ada tab; sebagai gantinya, gulir ke bawah ke bagian Penskalaan Ter jadwal.
5. Pilih tindakan terjadwal dan pilih Tindakan, Hapus.
6. Saat diminta konfirmasi, pilih Hapus.

Untuk mengelola penskalaan terjadwal menggunakan AWS CLI

Gunakan salah satu perintah berikut ini:

- put-scheduled-action
- describe-scheduled-actions
- delete-scheduled-action

# Pantau Armada EC2 atau Spot Fleet Anda

Pemantauan yang efektif terhadap Armada EC2 atau Spot Fleet Anda sangat penting untuk mempertahankan kinerja optimal dan memastikan keandalan. Ada berbagai alat untuk membantu Anda mencapai hal ini, termasuk Amazon CloudWatch dan Amazon EventBridge, yang dibahas dalam topik ini.

Dengan CloudWatch, Anda dapat mengumpulkan dan melacak metrik, mengatur alarm, dan secara otomatis bereaksi terhadap perubahan status armada Anda.

Dengan EventBridge, Anda dapat memantau dan merespons secara terprogram terhadap peristiwa yang dipancarkan oleh armada Anda. Dengan menentukan aturan di EventBridge, Anda dapat mengotomatiskan respons terhadap peristiwa armada tertentu, seperti penghentian instans atau perubahan status armada, meningkatkan efisiensi operasional Anda.

Topik

- Pantau Armada EC2 atau Spot Fleet Anda menggunakan CloudWatch

- Memantau dan merespons secara terprogram peristiwa yang dipancarkan oleh Armada EC2 atau Armada Spot menggunakan Amazon EventBridge

## Pantau Armada EC2 atau Spot Fleet Anda menggunakan CloudWatch

Anda dapat memantau Armada EC2 atau Armada Spot menggunakan CloudWatch metrik Amazon yang dijelaskan di bagian ini.

> **Important**
>
> Untuk memastikan keakuratannya, sebaiknya Anda mengaktifkan pemantauan mendetail saat menggunakan metrik ini. Untuk informasi selengkapnya, lihat Kelola pemantauan terperinci untuk instans EC2 Anda.

Untuk informasi lebih lanjut tentang penggunaan CloudWatch, lihatMemantau instans Anda menggunakan CloudWatch.

### Metrik Armada dan Spot Fleet EC2

`AWS/EC2Spot`Namespace mencakup metrik berikut untuk armada Anda, ditambah CloudWatch metrik untuk Instans Spot di armada Anda. Untuk informasi selengkapnya, lihat Metrik instans.

| Metrik | Deskripsi |
| --- | --- |
| `AvailableInstancePoolsCount` | Kumpulan kapasitas Spot yang ditentukan dalam permintaan armada.<br><br>Unit: Hitungan |
| `BidsSubmittedForCapacity` | Kapasitas di mana Amazon EC2 telah mengirimkan permintaan armada.<br><br>Unit: Hitungan |
| `EligibleInstancePoolCount` | Kumpulan kapasitas Spot yang ditentukan dalam permintaan armada tempat Amazon EC2 dapat memenuhi permintaan. Amazon EC2 tidak memenuhi permintaan di kumpulan di mana harga maksimum yang bersedia Anda bayar untuk |

| Metrik | Deskripsi |
| --- | --- |
| | Instans Spot kurang dari harga Spot atau harga Spot lebih besar dari harga On-Demand Instans.<br><br>Unit: Hitungan |
| `FulfilledCapacity` | Kapasitas yang telah dipenuhi oleh Amazon EC2.<br><br>Unit: Hitungan |
| `MaxPercentCapacityAllocation` | Nilai maksimum `PercentCapacityAllocation` di semua kumpulan armada yang ditentukan dalam permintaan armada.<br><br>Unit: Persen |
| `PendingCapacity` | Perbedaan antara `TargetCapacity` dan `FulfilledCapacity`.<br><br>Unit: Jumlah |
| `PercentCapacityAllocation` | Kapasitas yang dialokasikan untuk kolam kapasitas Spot untuk dimensi tertentu. Agar nilai maksimum dapat tercatat di semua kolam kapasitas Spot, gunakan `MaxPercentCapacityAllocation`.<br><br>Unit: Persen |
| `TargetCapacity` | Kapasitas target permintaan armada.<br><br>Unit: Hitungan |
| `TerminatingCapacity` | Kapasitas yang sedang diakhiri karena kapasitas yang disediakan lebih besar dari kapasitas target.<br><br>Unit: Jumlah |

Jika unit ukuran untuk metrik adalah `Count`, statistik yang paling berguna adalah `Average`.

## Dimensi Armada dan Spot Armada EC2

Untuk memfilter data armada Anda, gunakan dimensi berikut.

| Dimensi | Deskripsi |
| --- | --- |
| `AvailabilityZone` | Filter data berdasarkan Zona Ketersediaan. |
| `FleetRequestId` | Filter data berdasarkan permintaan armada. |
| `InstanceType` | Filter data menurut tipe instans. |

## Melihat CloudWatch metrik untuk Armada EC2 atau Armada Spot

Anda dapat melihat CloudWatch metrik untuk armada Anda menggunakan CloudWatch konsol Amazon. Metrik ini ditampilkan sebagai grafik pemantauan. Grafik ini menunjukkan titik data jika armada aktif.

Metrik dikelompokkan terlebih dahulu berdasarkan namespace, kemudian berdasarkan berbagai kombinasi dimensi di dalam setiap namespace. Misalnya, Anda dapat melihat semua metrik armada atau grup metrik armada berdasarkan ID permintaan armada, jenis instans, atau Zona Ketersediaan.

Untuk melihat metrik armada

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, perluas Metrik, dan pilih Semua metrik.
3. Pilih namespace Spot EC2.

> **Note**
>
> Jika namespace Spot EC2 tidak ditampilkan, ada dua alasan untuk ini. Entah Anda belum pernah menggunakan EC2 Fleet atau Spot Fleet di Wilayah tersebut—hanya AWS layanan yang Anda gunakan yang mengirim metrik ke Amazon. CloudWatch Atau, jika Anda telah menggunakan EC2 Fleet atau Spot Fleet di Wilayah, tetapi tidak selama dua minggu terakhir, namespace tidak muncul.

4. Untuk memfilter metrik berdasarkan dimensi, pilih salah satu dari berikut ini:

   - Metrik Permintaan Armada — Kelompokkan berdasarkan permintaan armada

- Berdasarkan Zona Ketersediaan — Kelompokkan berdasarkan permintaan armada dan Zona Ketersediaan
- Berdasarkan Jenis Instans — Kelompokkan berdasarkan permintaan armada dan jenis instans
- Berdasarkan Zone/Instance Jenis Ketersediaan — Kelompokkan berdasarkan permintaan armada, Zona Ketersediaan, dan jenis instans

5. Untuk melihat data untuk metrik, pilih kotak centang di sebelah metrik.

## Memantau dan merespons secara terprogram peristiwa yang dipancarkan oleh Armada EC2 atau Armada Spot menggunakan Amazon EventBridge

Ketika keadaan Armada EC2 atau Armada Spot berubah, ia akan mengeluarkan pemberitahuan. Pemberitahuan tersedia sebagai acara yang dikirim ke Amazon EventBridge (sebelumnya dikenal sebagai Amazon Ev CloudWatch ents). Peristiwa dipancarkan atas dasar upaya terbaik.

Anda dapat menggunakan Amazon EventBridge untuk membuat aturan yang memicu tindakan terprogram sebagai respons terhadap suatu peristiwa. Misalnya, Anda dapat membuat dua EventBridge aturan: satu dipicu saat status armada berubah, dan yang lain dipicu ketika instance dalam armada dihentikan. Dalam contoh ini, Anda dapat mengonfigurasi aturan pertama sehingga, jika status armada berubah, aturan memanggil topik SNS, mengirimkan pemberitahuan email kepada Anda. Anda dapat mengonfigurasi aturan kedua sehingga, jika instance dalam armada dihentikan, aturan tersebut memanggil fungsi Lambda untuk meluncurkan instance baru.

> **Note**
>
> Hanya armada tipe `maintain` dan `request` yang memancarkan peristiwa. Armada tipe `instant` tidak memancarkan peristiwa karena armada tipe tersebut mengirimkan permintaan satu kali sinkron, dan status armada segera diketahui dalam respons. Untuk menggunakan Amazon EventBridge untuk memantau peristiwa armada, jenis permintaan harus `maintain` atau`request`.

Untuk petunjuk tentang cara menggambarkan riwayat peristiwa armada, lihatJelaskan riwayat acara untuk Armada EC2 Anda.

Topik

- Buat EventBridge aturan Amazon untuk memantau peristiwa Armada EC2 atau Spot Fleet
- Jenis acara Armada EC2
- Tipe peristiwa Armada Spot

## Buat EventBridge aturan Amazon untuk memantau peristiwa Armada EC2 atau Spot Fleet

Ketika pemberitahuan perubahan status dipancarkan untuk Armada EC2 atau Armada Spot, pemberitahuan tersebut dikirim sebagai acara ke Amazon EventBridge sebagai file JSON. Jika EventBridge mendeteksi pola peristiwa yang cocok dengan pola yang ditentukan dalam aturan, EventBridge memanggil target (atau target) yang ditentukan dalam aturan.

Anda dapat menulis EventBridge aturan untuk mengotomatiskan tindakan berdasarkan pola peristiwa yang cocok.

Bidang berikut dalam acara membentuk pola peristiwa yang ditentukan dalam aturan:

`"source": "aws.ec2fleet"`

Mengidentifikasi bahwa peristiwa tersebut berasal dari Armada EC2.

`"detail-type": "EC2 Fleet State Change"`

Mengidentifikasi tipe peristiwa.

`"detail": { "sub-type": "submitted" }`

Mengidentifikasi subtipe peristiwa.

Untuk daftar peristiwa Armada dan Armada Spot EC2 dan contoh data peristiwa, lihat Jenis acara Armada EC2 danTipe peristiwa Armada Spot.

Contoh

- Membuat EventBridge aturan untuk mengirim notifikasi
- Buat EventBridge aturan untuk memicu fungsi Lambda

### Membuat EventBridge aturan untuk mengirim notifikasi

Contoh berikut membuat EventBridge aturan untuk mengirim email, pesan teks, atau pemberitahuan push seluler setiap kali Amazon EC2 memancarkan pemberitahuan perubahan status Armada EC2.

Sinyal dalam contoh ini dipancarkan sebagai peristiwa `EC2 Fleet State Change`, yang memicu tindakan yang ditentukan oleh aturan.

Prasyarat

Sebelum membuat EventBridge aturan, Anda harus membuat topik Amazon SNS untuk email, pesan teks, atau pemberitahuan push seluler.

Untuk membuat EventBridge aturan untuk mengirim pemberitahuan saat status Armada EC2 berubah

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Pilih Buat aturan.
3. Untuk Tentukan detail aturan, lakukan hal berikut:
   a. Masukkan Nama untuk aturan tersebut dan, secara opsional, deskripsi.

      Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah yang sama dan di bus peristiwa yang sama.
   b. Untuk Bus peristiwa, pilih default. Ketika AWS layanan di akun Anda menghasilkan acara, layanan selalu masuk ke bus acara default akun Anda.
   c. Untuk Tipe aturan, pilih Aturan dengan pola peristiwa.
   d. Pilih Selanjutnya.
4. Untuk Pola peristiwa build, lakukan hal berikut ini:
   a. Untuk Sumber acara, pilih AWS acara atau acara EventBridge mitra.
   b. Untuk Pola peristiwa, untuk contoh ini Anda akan menentukan pola peristiwa berikut agar sesuai dengan peristiwa `EC2 Fleet Instance Change`.

      ```
      {
       "source": ["aws.ec2fleet"],
       "detail-type": ["EC2 Fleet Instance Change"]
      }
      ```

      Untuk menambahkan pola peristiwa, Anda dapat menggunakan templat dengan memilih Bentuk pola peristiwa, atau menentukan pola Anda sendiri dengan memilih Pola kustom (editor JSON), sebagai berikut:

      i. Untuk menggunakan templat untuk membuat pola peristiwa, lakukan hal berikut:

A. Pilih Formulir pola peristiwa.

B. Untuk Sumber peristiwa, pilih Layanan AWS .

C. Untuk Layanan AWS , pilih Armada EC2.

D. Untuk Tipe peristiwa, pilih Perubahan Instans Armada EC2.

E. Untuk menyesuaikan templat, pilih Edit pola dan buat perubahan Anda agar sesuai dengan contoh pola peristiwa.

ii. (Alternatif) Untuk menentukan pola peristiwa khusus, lakukan hal berikut:

A. Pilih Pola kustom (editor JSON).

B. Dalam kotak Pola peristiwa, tambahkan pola peristiwa untuk contoh ini.

c. Pilih Selanjutnya.

5. Untuk Pilih target, lakukan hal berikut:

a. Untuk Tipe Target, pilih Layanan AWS .

b. Untuk Memilih target, pilih topik SNS untuk mengirim email, pesan teks, atau notifikasi push seluler saat peristiwa tersebut terjadi.

c. Untuk Topik, pilih topik yang ada. Anda harus terlebih dahulu membuat topik Amazon SNS menggunakan konsol Amazon SNS. Untuk informasi selengkapnya, lihat Menggunakan Amazon SNS untuk pengiriman pesan aplikasi-ke-orang (A2P) dalam Panduan Developer Amazon Simple Notification Service.

d. (Opsional) Pada Pengaturan tambahan, Anda dapat mengonfigurasi pengaturan tambahan secara opsional. Untuk informasi selengkapnya, lihat Membuat EventBridge aturan Amazon yang bereaksi terhadap peristiwa (langkah 16) di Panduan EventBridge Pengguna Amazon.

e. Pilih Selanjutnya.

6. (Opsional) Untuk Tanda, Anda dapat secara opsional menetapkan satu atau beberapa tanda ke aturan, lalu pilih Berikutnya.

7. Untuk Tinjau dan buat, lakukan hal berikut:

a. Tinjau detail aturan dan modifikasi seperlunya.

b. Pilih Buat aturan.

Untuk informasi selengkapnya, lihat EventBridge aturan Amazon dan pola EventBridge acara Amazon di Panduan EventBridge Pengguna Amazon

Buat EventBridge aturan untuk memicu fungsi Lambda

Contoh berikut membuat EventBridge aturan untuk memicu fungsi Lambda setiap kali Amazon EC2 memancarkan pemberitahuan perubahan instans EC2 Fleet saat instance diluncurkan. Sinyal dalam contoh ini dipancarkan sebagai peristiwa `EC2 Fleet Instance Change`, subtipe `launched`, yang memicu tindakan yang ditentukan oleh aturan.

Sebelum membuat EventBridge aturan, Anda harus membuat fungsi Lambda.

Untuk membuat fungsi Lambda untuk digunakan dalam aturan EventBridge

1. Buka AWS Lambda konsol di https://console.aws.amazon.com/lambda/.
2. Pilih Buat fungsi.
3. Masukkan nama untuk fungsi Anda, konfigurasikan kodenya, lalu pilih Buat fungsi.

   Untuk informasi selengkapnya, lihat Membuat fungsi Lambda pertama Anda di Panduan Peng AWS Lambda embang.

Untuk membuat EventBridge aturan untuk memicu fungsi Lambda saat instance di Armada EC2 berubah status

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Pilih Buat aturan.
3. Untuk Tentukan detail aturan, lakukan hal berikut:

   a. Masukkan Nama untuk aturan tersebut dan, secara opsional, deskripsi.

      Aturan tidak boleh memiliki nama yang sama dengan aturan lain di Wilayah yang sama dan di bus peristiwa yang sama.

   b. Untuk Bus peristiwa, pilih default. Ketika AWS layanan di akun Anda menghasilkan acara, layanan selalu masuk ke bus acara default akun Anda.

   c. Untuk Tipe aturan, pilih Aturan dengan pola peristiwa.

   d. Pilih Selanjutnya.

4. Untuk Pola peristiwa build, lakukan hal berikut ini:

a. Untuk Sumber acara, pilih AWS acara atau acara EventBridge mitra.

b. Untuk Pola peristiwa, untuk contoh ini Anda akan menentukan pola peristiwa berikut agar sesuai dengan peristiwa `EC2 Fleet Instance Change` dan subtipe `launched`.

```
{
 "source": ["aws.ec2fleet"],
 "detail-type": ["EC2 Fleet Instance Change"],
 "detail": {
   "sub-type": ["launched"]
}
```

Untuk menambahkan pola peristiwa, Anda dapat menggunakan templat dengan memilih Bentuk pola peristiwa, atau menentukan pola Anda sendiri dengan memilih Pola kustom (editor JSON), sebagai berikut:

i. Untuk menggunakan templat untuk membuat pola peristiwa, lakukan hal berikut:

A. Pilih Formulir pola peristiwa.

B. Untuk Sumber peristiwa, pilih Layanan AWS .

C. Untuk Layanan AWS , pilih Armada EC2.

D. Untuk Tipe peristiwa, pilih Perubahan Instans Armada EC2.

E. Pilih Edit pola, dan tambahkan `"detail": {"sub-type": ["launched"]` agar sesuai dengan contoh pola peristiwa. Untuk format JSON yang tepat, masukkan koma (`,`) setelah tanda kurung siku sebelumnya (`]`).

ii. (Alternatif) Untuk menentukan pola peristiwa kustom, lakukan hal berikut:

A. Pilih Pola kustom (editor JSON).

B. Dalam kotak Pola peristiwa, tambahkan pola peristiwa untuk contoh ini.

c. Pilih Selanjutnya.

5. Untuk Pilih target, lakukan hal berikut:

a. Untuk Tipe Target, pilih Layanan AWS .

b. Untuk Memilih target, pilih topik SNS untuk mengirim email, pesan teks, atau notifikasi push seluler saat peristiwa tersebut terjadi.

c. Untuk Topik, pilih fungsi Lambda, dan untuk Fungsi, pilih fungsi yang Anda buat untuk merespons saat peristiwa terjadi.

d. (Opsional) Di bawah Pengaturan tambahan, Anda dapat mengonfigurasi pengaturan tambahan secara opsional. Untuk informasi selengkapnya, lihat Membuat EventBridge aturan Amazon yang bereaksi terhadap peristiwa (langkah 16) di Panduan EventBridge Pengguna Amazon.

e. Pilih Selanjutnya.

6. (Opsional) Untuk Tanda, Anda dapat secara opsional menetapkan satu atau beberapa tanda ke aturan, lalu pilih Berikutnya.

7. Untuk Tinjau dan buat, lakukan hal berikut:

a. Tinjau detail aturan dan modifikasi seperlunya.

b. Pilih Buat aturan.

Untuk tutorial tentang cara membuat fungsi Lambda dan EventBridge aturan yang menjalankan fungsi Lambda, lihat Tutorial: Catat Status Menggunakan Instans Amazon EC2 EventBridge di Panduan Pengembang. AWS Lambda

## Jenis acara Armada EC2

Ada lima jenis acara Armada EC2. Untuk setiap jenis acara, ada beberapa sub-jenis.

Tipe peristiwa

- Perubahan Status Armada EC2
- Perubahan Permintaan Instans Spot Armada EC2
- Perubahan Instans Armada EC2
- Informasi Armada EC2
- Kesalahan Armada EC2

Perubahan Status Armada EC2

Armada EC2 mengirimkan `EC2 Fleet State Change` acara ke Amazon EventBridge ketika Armada EC2 berubah status.

Berikut adalah data contoh untuk peristiwa ini.

```
{
```

```
    "version": "0",
    "id": "715ed6b3-b8fc-27fe-fad6-528c7b8bf8a2",
    "detail-type": "EC2 Fleet State Change",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T09:00:20Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-598fb973-87b7-422d-
be4d-6b0809bfff0a"
    ],
    "detail": {
        "sub-type": "active"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`active`

Armada EC2 telah divalidasi dan Amazon EC2 berupaya untuk mempertahankan jumlah target dari instans yang sedang berjalan.

`deleted`

Permintaan Armada EC2 dihapus dan tidak ada instans yang berjalan. Armada EC2 akan dihapus dua hari setelah instansnya diakhiri.

`deleted_running`

Permintaan Armada EC2 dihapus dan tidak meluncurkan instans tambahan. Instans yang ada terus berjalan hingga diinterupsi atau diakhiri. Permintaan tetap berada dalam status ini sampai semua instans diinterupsi atau diakhiri.

`deleted_terminating`

Permintaan Armada EC2 dihapus dan instansnya diakhiri. Permintaan tetap berada dalam status ini sampai semua instans diakhiri.

`expired`

Permintaan Armada EC2 telah kedaluwarsa. Jika permintaan itu dibuat dengan set `TerminateInstancesWithExpiration`, peristiwa `terminated` berikutnya menunjukkan bahwa instans diakhiri.

`modify_in_progress`

Permintaan Armada EC2 sedang dimodifikasi. Permintaan tetap berada dalam status ini sampai modifikasi diproses sepenuhnya.

`modify_succeeded`

Permintaan Armada EC2 telah dimodifikasi.

`submitted`

Permintaan Armada EC2 sedang dievaluasi dan Amazon EC2 sedang bersiap untuk meluncurkan jumlah target instans.

`progress`

Permintaan Armada EC2 sedang dalam proses dipenuhi.

### Perubahan Permintaan Instans Spot Armada EC2

EC2 Fleet mengirimkan `EC2 Fleet Spot Instance Request Change` acara ke Amazon EventBridge saat permintaan Instans Spot di armada berubah status.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "19331f74-bf4b-a3dd-0f1b-ddb1422032b9",
    "detail-type": "EC2 Fleet Spot Instance Request Change",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T09:00:05Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/
fleet-83fd4e48-552a-40ef-9532-82a3acca5f10"
    ],
    "detail": {
        "spot-instance-request-id": "sir-rmqske6h",
        "description": "SpotInstanceRequestId sir-rmqske6h, PreviousState:
 cancelled_running",
        "sub-type": "cancelled"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`active`

Permintaan Instans Spot terpenuhi dan memiliki Instans Spot terkait.

`cancelled`

Anda membatalkan permintaan Instans Spot atau permintaan Instans Spot kedaluwarsa.

`disabled`

Anda menghentikan Instans Spot.

`submitted`

Permintaan Instans Spot dikirim.

### Perubahan Instans Armada EC2

EC2 Fleet mengirimkan`EC2 Fleet Instance Change` acara ke Amazon EventBridge ketika sebuah instance dalam armada berubah status.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "542ce428-c8f1-0608-c015-e8ed6522c5bc",
    "detail-type": "EC2 Fleet Instance Change",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T09:00:23Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-598fb973-87b7-422d-
be4d-6b0809bfff0a"
    ],
    "detail": {
        "instance-id": "i-0c594155dd5ff1829",
        "description": "{\"instanceType\":\"c5.large\",\"image\":\"ami-6057e21a\",
\"productDescription\":\"Linux/UNIX\",\"availabilityZone\":\"us-east-1d\"}",
        "sub-type": "launched"
    }
}
```

Nilai yang mungkin untuk`sub-type` adalah:

`launched`

Instans baru sudah diluncurkan.

`terminated`

Instans diakhiri.

`termination_notified`

Notifikasi pengakhiran instans dikirim ketika Instans Spot diakhiri oleh Amazon EC2 selama penurunan skala, ketika kapasitas target armada diturunkan, misalnya, dari kapasitas target 4 ke kapasitas target 3.

### Informasi Armada EC2

EC2 Fleet mengirimkan `EC2 Fleet Information` acara ke Amazon EventBridge ketika ada kesalahan selama pemenuhan. Peristiwa informasi tidak memblokir armada untuk mencoba memenuhi kapasitas targetnya.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "76529817-d605-4571-7224-d36cc1b2c0c4",
    "detail-type": "EC2 Fleet Information",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-11-09T08:17:07Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-8becf5fe-
bb9e-415d-8f54-3fa5a8628b91"
    ],
    "detail": {
        "description": "c4.xlarge, ami-0947d2ba12ee1ff75, Linux/UNIX, us-east-1a,
 Spot price in either SpotFleetRequestConfigData or SpotFleetLaunchSpecification or
 LaunchTemplate or LaunchTemplateOverrides is less than Spot market price $0.0619",
        "sub-type": "launchSpecUnusable"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`fleetProgressHalted`

Harga di setiap spesifikasi peluncuran tidak berlaku karena berada di bawah harga Spot (semua spesifikasi peluncuran telah menghasilkan peristiwa `launchSpecUnusable`). Spesifikasi peluncuran mungkin menjadi valid jika harga Spot berubah.

`launchSpecTemporarilyBlacklisted`

Konfigurasi tidak valid dan beberapa upaya untuk meluncurkan instans gagal. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

`launchSpecUnusable`

Harga dalam spesifikasi peluncuran tidak berlaku karena berada di bawah harga Spot.

`registerWithLoadBalancersFailed`

Upaya untuk mendaftarkan instans dengan penyeimbang beban gagal. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

### Kesalahan Armada EC2

EC2 Fleet mengirimkan `EC2 Fleet Error` acara ke Amazon EventBridge ketika ada kesalahan selama pemenuhan. Peristiwa kesalahan memblokir armada untuk mencoba memenuhi kapasitas targetnya.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "69849a22-6d0f-d4ce-602b-b47c1c98240e",
    "detail-type": "EC2 Fleet Error",
    "source": "aws.ec2fleet",
    "account": "123456789012",
    "time": "2020-10-07T01:44:24Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:fleet/fleet-9bb19bc6-60d3-4fd2-ae47-
d33e68eafa08"
    ],
    "detail": {
```

```
        "description": "m3.large, ami-00068cd7555f543d5, Linux/UNIX: IPv6 is not
 supported for the instance type 'm3.large'. ",
        "sub-type": "spotFleetRequestConfigurationInvalid"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`iamFleetRoleInvalid`

Armada EC2 tidak memiliki izin yang diperlukan untuk meluncurkan atau mengakhiri instans.

`allLaunchSpecsTemporarilyBlacklisted`

Tidak ada konfigurasi yang valid, dan beberapa upaya untuk meluncurkan instans gagal. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

`spotInstanceCountLimitExceeded`

Anda telah mencapai batas jumlah Instans Spot yang dapat diluncurkan.

`spotFleetRequestConfigurationInvalid`

Konfigurasi tidak valid. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

## Tipe peristiwa Armada Spot

> **Note**
>
> Spot Fleet menggunakan API lama tanpa investasi yang direncanakan. Sebaiknya gunakan EC2 Fleet atau grup Auto Scaling sebagai gantinya. Untuk informasi selengkapnya, lihat Metode armada mana yang terbaik untuk digunakan?.

Ada lima tipe peristiwa Armada Spot. Untuk setiap tipe peristiwa, ada beberapa subtipe.

Tipe peristiwa

- Perubahan Status Armada Spot EC2
- Perubahan Permintaan Instans Spot Armada Spot EC2
- Perubahan Instans Armada Spot EC2
- Informasi Armada Spot EC2

- Kesalahan Armada Spot EC2

Perubahan Status Armada Spot EC2

Spot Fleet mengirimkan file`EC2 Spot Fleet State Change` acara ke Amazon EventBridge saat Fleet Spot berubah status.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "d1af1091-6cc3-2e24-203a-3b870e455d5b",
    "detail-type": "EC2 Spot Fleet State Change",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T08:57:06Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-4b6d274d-0cea-4b2c-
b3be-9dc627ad1f55"
    ],
    "detail": {
        "sub-type": "submitted"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`active`

Permintaan Armada Spot telah divalidasi dan Amazon EC2 berupaya untuk mempertahankan jumlah target instans yang sedang berjalan.

`cancelled`

Permintaan Armada Spot dibatalkan dan tidak ada instans yang berjalan. Armada Spot akan dihapus dua hari setelah instansnya diakhiri.

`cancelled_running`

Permintaan Armada Spot dibatalkan dan tidak meluncurkan instans tambahan. Instans yang ada terus berjalan hingga diinterupsi atau diakhiri. Permintaan tetap berada dalam status ini sampai semua instans diinterupsi atau diakhiri.

`cancelled_terminating`

Permintaan Armada Spot dibatalkan dan instansnya diakhiri. Permintaan tetap berada dalam status ini sampai semua instans diakhiri.

`expired`

Permintaan Armada Spot telah kedaluwarsa. Jika permintaan itu dibuat dengan set `TerminateInstancesWithExpiration`, peristiwa `terminated` berikutnya menunjukkan bahwa instans diakhiri.

`modify_in_progress`

Permintaan Armada Spot sedang dimodifikasi. Permintaan tetap berada dalam status ini sampai modifikasi diproses sepenuhnya.

`modify_succeeded`

Permintaan Armada Spot telah dimodifikasi.

`submitted`

Permintaan Armada Spot sedang dievaluasi dan Amazon EC2 sedang bersiap untuk meluncurkan jumlah target instans.

`progress`

Permintaan Armada Spot sedang dalam proses dipenuhi.

### Perubahan Permintaan Instans Spot Armada Spot EC2

Spot Fleet mengirimkan `EC2 Spot Fleet Spot Instance Request Change` acara ke Amazon EventBridge saat permintaan Instans Spot di armada berubah status.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "cd141ef0-14af-d670-a71d-fe46e9971bd2",
    "detail-type": "EC2 Spot Fleet Spot Instance Request Change",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T08:53:21Z",
    "region": "us-east-1",
    "resources": [
```

```
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-
a98d2133-941a-47dc-8b03-0f94c6852ad1"
    ],
    "detail": {
        "spot-instance-request-id": "sir-a2w9gc5h",
        "description": "SpotInstanceRequestId sir-a2w9gc5h, PreviousState:
 cancelled_running",
        "sub-type": "cancelled"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`active`

Permintaan Instans Spot terpenuhi dan memiliki Instans Spot terkait.

`cancelled`

Anda membatalkan permintaan Instans Spot atau permintaan Instans Spot kedaluwarsa.

`disabled`

Anda menghentikan Instans Spot.

`submitted`

Permintaan Instans Spot dikirim.

### Perubahan Instans Armada Spot EC2

Spot Fleet mengirimkan `EC2 Spot Fleet Instance Change` acara ke Amazon EventBridge saat instance di armada berubah status.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "11591686-5bd7-bbaa-eb40-d46529c2710f",
    "detail-type": "EC2 Spot Fleet Instance Change",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T07:25:02Z",
    "region": "us-east-1",
    "resources": [
```

```
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-c8a764a4-bedc-4b62-
af9c-0095e6e3ba61"
    ],
    "detail": {
        "instance-id": "i-08b90df1e09c30c9b",
        "description": "{\"instanceType\":\"r4.2xlarge\",\"image\":
\"ami-032930428bf1abbff\",\"productDescription\":\"Linux/UNIX\",\"availabilityZone\":
\"us-east-1a\"}",
        "sub-type": "launched"
    }
}
```

Nilai yang mungkin untuk`sub-type` adalah:

`launched`

Instans baru sudah diluncurkan.

`terminated`

Instans diakhiri.

`termination_notified`

Notifikasi pengakhiran instans dikirim ketika Instans Spot diakhiri oleh Amazon EC2 selama penurunan skala, ketika kapasitas target armada diturunkan, misalnya, dari kapasitas target 4 ke kapasitas target 3.

### Informasi Armada Spot EC2

Spot Fleet mengirimkan file`EC2 Spot Fleet Information` acara ke Amazon EventBridge ketika ada kesalahan selama pemenuhan. Peristiwa informasi tidak memblokir armada untuk mencoba memenuhi kapasitas targetnya.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "73a60f70-3409-a66c-635c-7f66c5f5b669",
    "detail-type": "EC2 Spot Fleet Information",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-08T20:56:12Z",
    "region": "us-east-1",
```

```
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/sfr-2531ea06-
af18-4647-8757-7d69c94971b1"
    ],
    "detail": {
        "description": "r3.8xlarge, ami-032930428bf1abbff, Linux/UNIX, us-east-1a, Spot
 bid price is less than Spot market price $0.5291",
        "sub-type": "launchSpecUnusable"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`fleetProgressHalted`

Harga di setiap spesifikasi peluncuran tidak berlaku karena berada di bawah harga Spot (semua spesifikasi peluncuran telah menghasilkan peristiwa `launchSpecUnusable`). Spesifikasi peluncuran mungkin menjadi valid jika harga Spot berubah.

`launchSpecTemporarilyBlacklisted`

Konfigurasi tidak valid dan beberapa upaya untuk meluncurkan instans gagal. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

`launchSpecUnusable`

Harga dalam spesifikasi peluncuran tidak berlaku karena berada di bawah harga Spot.

`registerWithLoadBalancersFailed`

Upaya untuk mendaftarkan instans dengan penyeimbang beban gagal. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

### Kesalahan Armada Spot EC2

Spot Fleet mengirimkan `EC2 Spot Fleet Error` acara ke Amazon EventBridge ketika ada kesalahan selama pemenuhan. Peristiwa kesalahan memblokir armada untuk mencoba memenuhi kapasitas targetnya.

Berikut adalah data contoh untuk peristiwa ini.

```
{
    "version": "0",
    "id": "10adc4e7-675c-643e-125c-5bfa1b1ba5d2",
```

```
    "detail-type": "EC2 Spot Fleet Error",
    "source": "aws.ec2spotfleet",
    "account": "123456789012",
    "time": "2020-11-09T06:56:07Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:ec2:us-east-1:123456789012:spot-fleet-request/
sfr-38725d30-25f1-4f30-83ce-2907c56dba17"
    ],
    "detail": {
        "description": "r4.2xlarge, ami-032930428bf1abbff, Linux/UNIX: The
 associatePublicIPAddress parameter can only be specified for the network interface
 with DeviceIndex 0. ",
        "sub-type": "spotFleetRequestConfigurationInvalid"
    }
}
```

Nilai yang mungkin untuk `sub-type` adalah:

`iamFleetRoleInvalid`

Armada Spot tidak memiliki izin yang diperlukan untuk meluncurkan atau mengakhiri sebuah instans.

`allLaunchSpecsTemporarilyBlacklisted`

Tidak ada konfigurasi yang valid, dan beberapa upaya untuk meluncurkan instans gagal. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

`spotInstanceCountLimitExceeded`

Anda telah mencapai batas jumlah Instans Spot yang dapat diluncurkan.

`spotFleetRequestConfigurationInvalid`

Konfigurasi tidak valid. Untuk informasi selengkapnya, lihat deskripsi peristiwa.

# Tutorial untuk Armada EC2

Ada berbagai cara untuk mengkonfigurasi Armada EC2. Konfigurasi yang Anda pilih tergantung pada kasus penggunaan spesifik Anda.

Tutorial berikut mencakup beberapa kasus penggunaan yang mungkin dan menyediakan tugas yang diperlukan untuk mengimplementasikannya.

| Kasus penggunaan | Tautan ke tutorial |
| --- | --- |
| Gunakan pembobotan instans untuk mengelola ketersediaan dan kinerja Armada EC2 Anda.<br><br>Dengan bobot instans, Anda menetapkan bobot untuk setiap jenis instans di Armada EC2 Anda untuk mewakili kapasitas komputasi dan kinerja relatif satu sama lain. Berdasarkan bobot, armada dapat menggunakan kombinasi jenis instans yang ditentukan, selama dapat memenuhi kapasitas target yang diinginkan. | Tutorial: Mengkonfigurasi Armada EC2 untuk menggunakan pembobotan instans |
| Gunakan On-Demand kapasitas untuk memastikan ketersediaan selama periode puncak, tetapi manfaatkan kapasitas Spot tambahan dengan biaya lebih rendah.<br><br>Konfigurasikan Armada EC2 Anda untuk menggunakan On-Demand Instans sebagai kapasitas utama untuk memastikan kapasitas yang tersedia selama periode puncak. Selain itu, alokasikan beberapa kapasitas ke Instans Spot untuk mendapatkan keuntungan dari harga diskon, namun perlu diingat bahwa Instans Spot dapat terganggu jika Amazon EC2 membutuhkan kapasitas kembali. | Tutorial: Mengkonfigurasi Armada EC2 untuk menggunakan On-Demand Instans sebagai kapasitas utama |
| Gunakan Reservasi Kapasitas untuk mencadangkan kapasitas komputasi untuk On-Demand Instans Anda.<br><br>Konfigurasikan Armada EC2 Anda untuk menggunakan Reservasi `targeted` Kapasitas terlebih dahulu saat meluncurkan On-Demand Instans. Jika Anda memiliki persyaratan kapasitas yang ketat, dan menjalankan beban kerja penting bisnis yang memerlukan | Tutorial: Mengkonfigurasi Armada EC2 untuk meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan |

| Kasus penggunaan | Tautan ke tutorial |
| --- | --- |
| jaminan kapasitas pada tingkat jangka panjang atau jangka pendek tertentu, kami sarankan Anda membuat Reservasi Kapasitas untuk memastikan bahwa Anda selalu memiliki akses ke kapasitas Amazon EC2 saat Anda membutuhkannya, selama Anda membutuhk annya. | |
| Gunakan Blok Kapasitas untuk memesan instans GPU yang sangat dicari untuk beban kerja ML Anda.<br><br>Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Blok Kapasitas. | Tutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Blok Kapasitas |
| Gunakan Reservasi Kapasitas Terputus untuk sementara menggunakan kembali kapasitas idle di seluruh AWS Organisasi Anda dan menghemat biaya.<br><br>Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Reservasi Kapasitas Terputus, yang mewakili kapasitas cadangan dalam organisasi Anda AWS . Pemilik kapasitas dapat merebut kembali kapasitas kapan saja. Setelah direklamasi, EC2 mengakhiri instans setelah pemberitahuan 2 menit. | Tutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Reservasi Kapasitas yang Dapat Diinterupsi |

| Kasus penggunaan | Tautan ke tutorial |
| --- | --- |
| Gunakan Grup Sumber Daya Reservasi Kapasitas tunggal untuk meluncurkan instans ke beberapa jenis Reservasi Kapasitas.<br><br>Konfigurasikan Armada EC2 Anda untuk meluncurkan instans di beberapa jenis Reservasi Kapasitas — Reservasi On-Demand Kapasitas, Blok Kapasitas untuk ML, dan Reservasi Kapasitas yang dapat diinterupsi — menggunakan Grup Sumber Daya Reservasi Kapasitas tunggal. Anda dapat mempriori taskan jenis Reservasi Kapasitas dalam urutan tertentu, dan secara opsional kembali ke On-Demand kapasitas ketika kapasitas yang dipesan tidak mencukupi untuk memenuhi target Anda. | Tutorial: Mengkonfigurasi Armada EC2 Anda untuk meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas |

# Tutorial: Mengkonfigurasi Armada EC2 untuk menggunakan pembobotan instans

Tutorial ini menggunakan perusahaan fiktif bernama Example Corp untuk mengilustrasikan proses permintaan Armada EC2 menggunakan pembobotan instans.

## Tujuan

Example Corp, sebuah perusahaan farmasi, ingin menggunakan kekuatan komputasi Amazon EC2 untuk melakukan skrining senyawa kimia yang dapat digunakan untuk melawan kanker.

## Perencanaan

Pertama-tama, Example Corp meninjau Praktik Terbaik Spot. Selanjutnya, Example Corp menentukan kebutuhan untuk Armada EC2 mereka.

Jenis instance

Example Corp memiliki aplikasi intensif komputasi dan intensif memori yang memiliki performa terbaik dengan setidaknya 60 GB memori dan delapan virtual CPU (vCPU). Mereka ingin memaksimalkan sumber daya ini untuk aplikasi dengan harga serendah mungkin. Example Corp memutuskan bahwa tipe instans EC2 berikut akan memenuhi kebutuhan mereka:

| Jenis instans | Memori (GiB) | vCPU |
|---|---|---|
| r3.2xlarge | 61 | 8 |
| r3.4xlarge | 122 | 16 |
| r3.8xlarge | 244 | 32 |

Kapasitas target dalam unit

Dengan bobot instans, kapasitas target dapat sama dengan jumlah instance (default) atau kombinasi faktor seperti core (vCPU), memori (GiBs), dan penyimpanan (GB). Dengan mempertimbangkan dasar untuk aplikasi mereka (60 GB RAM dan delapan vCPU) sebagai satu unit, Example Corp memutuskan bahwa 20 kali jumlah ini akan memenuhi kebutuhan mereka. Jadi perusahaan menetapkan target kapasitas permintaan Armada EC2 mereka menjadi 20 unit.

Bobot instans

Setelah menentukan kapasitas target, Example Corp menghitung bobot instans. Guna menghitung bobot instans untuk setiap tipe instans, mereka menentukan unit dari setiap tipe instans yang diperlukan untuk mencapai kapasitas target sebagai berikut:

- r3.2xlarge (61,0 GB, 8 vCPU) = 1 unit dari 20
- r3.4xlarge (122,0 GB, 16 vCPU) = 2 unit dari 20
- r3.8xlarge (244,0 GB, 32 vCPU) = 4 unit dari 20

Oleh karena itu, Example Corp menetapkan bobot instans 1, 2, dan 4 ke konfigurasi peluncuran masing-masing dalam permintaan Armada EC2.

Harga per unit jam

Example Corp menggunakan On-Demand harga per instans jam sebagai titik awal untuk harganya. Mereka juga dapat menggunakan harga Spot baru-baru ini, atau kombinasi keduanya. Untuk menghitung harga per unit jam, mereka membagi harga awal per jam instans berdasarkan bobot. Misalnya:

| Tipe instans | On-Demand harga | Bobot instans | Harga per unit jam |
|---|---|---|---|
| r3.2xLarge | $0,7 | 1 | $0,7 |
| r3.4xLarge | $1,4 | 2 | $0,7 |
| r3.8xlarge | $2,8 | 4 | $0,7 |

Example Corp dapat menggunakan harga global per unit jam sebesar 0,7 USD dan kompetitif untuk ketiga tipe instans. Mereka juga dapat menggunakan harga global per unit jam 0,7 USD dan harga spesifik per unit jam 0,9 USD di spesifikasi peluncuran `r3.8xlarge`.

### Memverifikasi izin

Sebelum membuat Armada EC2, Example Corp memverifikasi bahwa mereka memiliki peran IAM dengan izin yang diperlukan. Untuk informasi selengkapnya, lihat Prasyarat Armada EC2.

### Membuat templat peluncuran

Selanjutnya, Example Corp membuat templat peluncuran. ID templat peluncuran digunakan di langkah berikut. Untuk informasi selengkapnya, lihat Buat template peluncuran Amazon EC2.

### Membuat Armada EC2

Example Corp membuat file, `config.json`, dengan konfigurasi sebagai berikut untuk Armada EC2 file tersebut. Dalam contoh berikut, ganti pengidentifikasi sumber daya dengan pengidentifikasi sumber daya Anda sendiri.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-07b3bc7625cdab851",
```

```
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "r3.2xlarge",
                    "SubnetId": "subnet-482e4972",
                    "WeightedCapacity": 1
                },
                {
                    "InstanceType": "r3.4xlarge",
                    "SubnetId": "subnet-482e4972",
                    "WeightedCapacity": 2
                },
                {
                    "InstanceType": "r3.8xlarge",
                    "MaxPrice": "0.90",
                    "SubnetId": "subnet-482e4972",
                    "WeightedCapacity": 4
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "DefaultTargetCapacityType": "spot"
    }
}
```

Example Corp membuat Armada EC2 menggunakan perintah create-fleet berikut.

```
aws ec2 create-fleet --cli-input-json file://config.json
```

Untuk informasi selengkapnya, lihat Membuat Armada EC2.

## Pemenuhan

Strategi alokasi menentukan asal dari kolam kapasitas Spot yang menjadi sumber Instans Spot Anda.

Dengan strategi `lowest-price` (yang merupakan strategi default), Instans Spot berasal dari kolam dengan harga terendah per unit pada saat pemenuhan. Untuk menyediakan kapasitas 20 unit, Armada EC2 meluncurkan 20 instans `r3.2xlarge` (20 dibagi 1), 10 instans `r3.4xlarge` (20 dibagi 2), atau 5 instans `r3.8xlarge` (20 dibagi 4).

Jika Example Corp menggunakan strategi `diversified`, Instans Spot akan berasal dari ketiga kolam. Armada EC2 akan meluncurkan 6 instans `r3.2xlarge` (yang menyediakan 6 unit), 3 instans `r3.4xlarge` (yang menyediakan 6 unit), dan 2 instans `r3.8xlarge` (yang menyediakan 8 unit), dengan total 20 unit.

# Tutorial: Mengkonfigurasi Armada EC2 untuk menggunakan On-Demand Instans sebagai kapasitas utama

Tutorial ini menggunakan perusahaan fiktif bernama ABC Online untuk menggambarkan proses meminta Armada EC2 dengan On-Demand kapasitas utama, dan kapasitas Spot jika tersedia.

## Tujuan

ABC Online, perusahaan pengiriman restoran, bertujuan untuk menyediakan kapasitas Amazon EC2 di seluruh jenis instans EC2 dan opsi pembelian untuk mencapai skala, kinerja, dan biaya yang diinginkan.

## Rencana

ABC Online membutuhkan kapasitas tetap untuk menangani periode puncak, tetapi ingin mendapatkan keuntungan dari kapasitas tambahan dengan biaya lebih rendah. Perusahaan menentukan persyaratan berikut untuk Armada EC2 mereka:

- On-Demand Kapasitas instans — ABC Online memerlukan 15 On-Demand Instans untuk memastikan bahwa mereka dapat mengakomodasi lalu lintas pada periode puncak.
- Kapasitas Instans Spot — Untuk meningkatkan kinerja, tetapi dengan harga yang lebih rendah, ABC Online berencana menyediakan 5 Instans Spot.

## Memverifikasi izin

Sebelum membuat Armada EC2, ABC Online memverifikasi bahwa mereka memiliki peran IAM dengan izin yang diperlukan. Untuk informasi selengkapnya, lihat Prasyarat Armada EC2.

## Membuat templat peluncuran

Selanjutnya, ABC Online membuat templat peluncuran. ID templat peluncuran digunakan di langkah berikut. Untuk informasi selengkapnya, lihat Buat template peluncuran Amazon EC2.

### Membuat Armada EC2

ABC Online membuat file, `config.json`, dengan konfigurasi sebagai berikut untuk Armada EC2-nya. Dalam contoh berikut, ganti pengidentifikasi sumber daya dengan pengidentifikasi sumber daya Anda sendiri.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-07b3bc7625cdab851",
                "Version": "2"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 20,
        "OnDemandTargetCapacity":15,
        "DefaultTargetCapacityType": "spot"
    }
}
```

ABC Online membuat Armada EC2 menggunakan perintah create-fleet berikut.

```
aws ec2 create-fleet --cli-input-json file://config.json
```

Untuk informasi selengkapnya, lihat Membuat Armada EC2.

### Pemenuhan

Strategi alokasi menentukan bahwa On-Demand kapasitas selalu terpenuhi, sedangkan keseimbangan kapasitas target terpenuhi sebagai Spot jika ada kapasitas yang tersedia.

## Tutorial: Mengkonfigurasi Armada EC2 untuk meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan

Tutorial ini memandu Anda melalui semua langkah yang harus Anda lakukan agar Armada EC2 Anda meluncurkan On-Demand Instans ke Reservasi `targeted` Kapasitas.

Anda akan mempelajari cara mengonfigurasi armada untuk menggunakan Reservasi `targeted` On-Demand Kapasitas terlebih dahulu saat meluncurkan On-Demand Instans. Anda juga akan mempelajari cara mengonfigurasi armada sehingga, ketika total kapasitas On-Demand target melebihi jumlah Reservasi Kapasitas yang tidak terpakai yang tersedia, armada menggunakan strategi alokasi yang ditentukan untuk memilih kumpulan instans untuk meluncurkan kapasitas target yang tersisa.

Konfigurasi Armada EC2

Dalam tutorial ini, armada dikonfigurasi sebagai berikut:

- Kapasitas target: 10 On-Demand Instans
- Total Reserv `targeted` asi Kapasitas yang tidak digunakan: 6 (kurang dari kapasitas On-Demand target armada 10 On-Demand Instans)
- Jumlah kolam Reservasi Kapasitas: 2 (`us-east-1a` dan `us-east-1b`)
- Jumlah Reservasi Kapasitas per kolam: 3
- On-Demand strategi alokasi: `lowest-price` (Jika jumlah Reservasi Kapasitas yang tidak digunakan kurang dari kapasitas On-Demand target, armada menentukan kumpulan tempat untuk meluncurkan On-Demand kapasitas yang tersisa berdasarkan strategi al On-Demand okasi.)

  Perhatikan bahwa Anda juga dapat menggunakan strategi alokasi `prioritized` alih-alih strategi alokasi `lowest-price`.

Untuk meluncurkan On-Demand Instans ke Reservasi targeted Kapasitas, Anda harus melakukan sejumlah langkah, sebagai berikut:

- Langkah 1: Membuat Reservasi Kapasitas
- Langkah 2: Membuat grup sumber daya Reservasi Kapasitas
- Langkah 3: Menambahkan Reservasi Kapasitas ke grup sumber daya Reservasi Kapasitas
- (Opsional) Langkah 4: Melihat Reservasi Kapasitas di grup sumber daya
- Langkah 5: Membuat templat peluncuran yang menentukan bahwa Reservasi Kapasitas menargetkan grup sumber daya tertentu
- (Opsional) Langkah 6: Mendeskripsikan templat peluncuran
- Langkah 7: Membuat Armada EC2
- (Opsional) Langkah 8: Melihat jumlah Reservasi Kapasitas yang tidak terpakai yang tersisa

## Langkah 1: Membuat Reservasi Kapasitas

Gunakan perintah create-capacity-reservation untuk membuat Reservasi Kapasitas, tiga untuk `us-east-1a` dan tiga lainnya untuk `us-east-1b`. Kecuali untuk Zona Ketersediaan, atribut lain dari Reservasi Kapasitas bersifat identik.

3 Reservasi Kapasitas di **`us-east-1a`**

```
aws ec2 create-capacity-reservation \
    --availability-zone us-east-1a \
    --instance-type c5.xlarge \
    --instance-platform Linux/UNIX \
    --instance-count 3 \
    --instance-match-criteria targeted
```

Contoh ID Reservasi Kapasitas yang dihasilkan

```
cr-1234567890abcdef1
```

3 Reservasi Kapasitas di **`us-east-1b`**

```
aws ec2 create-capacity-reservation \
    --availability-zone us-east-1b \
    --instance-type c5.xlarge \
    --instance-platform Linux/UNIX \
    --instance-count 3 \
    --instance-match-criteria targeted
```

Contoh ID Reservasi Kapasitas yang dihasilkan

```
cr-54321abcdef567890
```

## Langkah 2: Membuat grup sumber daya Reservasi Kapasitas

Gunakan layanan `resource-groups` dan perintah create-group untuk membuat grup sumber daya Reservasi Kapasitas. Dalam contoh ini, grup sumber daya diberi nama `my-cr-group`. Untuk informasi tentang alasan Anda harus membuat grup sumber daya, lihat Gunakan Reservasi Kapasitas untuk cadangan On-Demand kapasitas di Armada EC2.

```
aws resource-groups create-group \
```

```
    --name my-cr-group \
    --configuration '{"Type":"AWS::EC2::CapacityReservationPool"}'
 '{"Type":"AWS::ResourceGroups::Generic", "Parameters": [{"Name": "allowed-resource-
types", "Values": ["AWS::EC2::CapacityReservation"]}]}'
```

## Langkah 3: Menambahkan Reservasi Kapasitas ke grup sumber daya Reservasi Kapasitas

Gunakan layanan `resource-groups` dan perintah group-resources untuk menambahkan Reservasi Kapasitas yang Anda buat di Langkah 1 ke grup sumber daya Reservasi Kapasitas. Perhatikan bahwa Anda harus mereferensikan On-Demand Reservasi Kapasitas dengan ARN-nya.

```
aws resource-groups group-resources \
    --group my-cr-group \
    --resource-arns \
      arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1 \
      arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-54321abcdef567890
```

Contoh Output

```
{
   "Failed": [],
   "Succeeded": [
   "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-1234567890abcdef1",
   "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/cr-54321abcdef567890"
   ]
}
```

## (Opsional) Langkah 4: Melihat Reservasi Kapasitas di grup sumber daya

Gunakan layanan`resource-groups` dan perintah list-group-resources untuk mendeskripsikan grup sumber daya secara opsional guna melihat Reservasi Kapasitas.

```
aws resource-groups list-group-resources --group my-cr-group
```

Contoh output

```
{
    "ResourceIdentifiers": [
        {
            "ResourceType": "AWS::EC2::CapacityReservation",
```

```
            "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/
cr-1234567890abcdef1"
        },
        {
            "ResourceType": "AWS::EC2::CapacityReservation",
            "ResourceArn": "arn:aws:ec2:us-east-1:123456789012:capacity-reservation/
cr-54321abcdef567890"
        }
    ]
}
```

## Langkah 5: Membuat templat peluncuran yang menentukan bahwa Reservasi Kapasitas menargetkan grup sumber daya tertentu

Gunakan perintah create-launch-template untuk membuat templat peluncuran guna menentukan Reservasi Kapasitas yang akan digunakan. Dalam contoh ini, armada akan menggunakan Reservasi Kapasitas `targeted`, yang telah ditambahkan ke grup sumber daya. Oleh karena itu, data templat peluncuran menentukan bahwa Reservasi Kapasitas menargetkan grup sumber daya tertentu. Dalam contoh ini, templat peluncuran diberi nama `my-launch-template`.

```
aws ec2 create-launch-template \
    --launch-template-name my-launch-template \
    --launch-template-data \
        '{"ImageId": "ami-0123456789example",
          "CapacityReservationSpecification":
            {"CapacityReservationTarget":
                { "CapacityReservationResourceGroupArn": "arn:aws:resource-groups:us-
east-1:123456789012:group/my-cr-group" }
            }
        }'
```

## (Opsional) Langkah 6: Mendeskripsikan templat peluncuran

Gunakan perintah describe-launch-template-versions untuk mendeskripsikan template peluncuran secara opsional untuk melihat konfigurasinya.

```
aws ec2 describe-launch-template-versions --launch-template-name my-launch-template
```

Contoh Output

```
{
```

```
    "LaunchTemplateVersions": [
        {
            "LaunchTemplateId": "lt-01234567890example",
            "LaunchTemplateName": "my-launch-template",
            "VersionNumber": 1,
            "CreateTime": "2021-01-19T20:50:19.000Z",
            "CreatedBy": "arn:aws:iam::123456789012:user/Admin",
            "DefaultVersion": true,
            "LaunchTemplateData": {
                "ImageId": "ami-0947d2ba12ee1ff75",
                "CapacityReservationSpecification": {
                    "CapacityReservationTarget": {
                        "CapacityReservationResourceGroupArn": "arn:aws:resource-
groups:us-east-1:123456789012:group/my-cr-group"
                    }
                }
            }
        }
    ]
}
```

## Langkah 7: Membuat Armada EC2

Buat Armada EC2 yang menentukan informasi konfigurasi untuk instans yang akan diluncurkan. Konfigurasi Armada EC2 berikut hanya menampilkan konfigurasi terkait untuk contoh ini. Templat peluncuran `my-launch-template` adalah templat peluncuran yang Anda buat di Langkah 5. Terdapat dua kolam instans, masing-masing dengan tipe instans yang sama (`c5.xlarge`), tetapi dengan Zona Ketersediaan (`us-east-1a` dan `us-east-1b`) yang berbeda. Harga kolam instans sama karena harga ditentukan untuk Wilayah, bukan per Zona Ketersediaan. Total kapasitas target adalah 10, dan tipe kapasitas target default adalah `on-demand`. Strategi On-Demand alokasi adalah`lowest-price`. Strategi penggunaan untuk Reservasi Kapasitas adalah `use-capacity-reservations-first`.

> **Note**
>
> Tipe armada harus `instant`. Tipe armada lainnya tidak mendukung `use-capacity-reservations-first`.

```
{
    "LaunchTemplateConfigs": [
```

```
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
            "Overrides": [
               {
                   "InstanceType": "c5.xlarge",
                   "AvailabilityZone": "us-east-1a"
               },
               {
                    "InstanceType": "c5.xlarge",
                    "AvailabilityZone": "us-east-1b"
               }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 10,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions": {
             "UsageStrategy": "use-capacity-reservations-first"
         }
    },
    "Type": "instant"
}
```

Setelah Anda membuat armada `instant` menggunakan konfigurasi sebelumnya, 10 instans berikut diluncurkan untuk memenuhi kapasitas target:

- Reservasi Kapasitas digunakan terlebih dahulu untuk meluncurkan 6 On-Demand Instans sebagai berikut:
  - 3 On-Demand Instans diluncurkan ke 3 Reservasi `c5.xlarge` `targeted` Kapasitas di `us-east-1a`
  - 3 On-Demand Instans diluncurkan ke 3 Reservasi `c5.xlarge` `targeted` Kapasitas di `us-east-1b`
- Untuk memenuhi kapasitas target, 4 On-Demand Instans tambahan diluncurkan ke On-Demand kapasitas reguler sesuai dengan strategi On-Demand alokasi, yang ada `lowest-price` dalam

contoh ini. Namun, karena kumpulan memiliki harga yang sama (karena harga per Wilayah dan bukan per Zona Ketersediaan), armada meluncurkan 4 On-Demand Instans yang tersisa ke salah satu kumpulan.

### (Opsional) Langkah 8: Melihat jumlah Reservasi Kapasitas yang tidak terpakai yang tersisa

Setelah armada diluncurkan, Anda dapat menjalankan describe-capacity-reservations untuk melihat banyaknya Reservasi Kapasitas yang tidak terpakai yang tersisa. Dalam contoh ini, Anda akan melihat respons berikut, yang menunjukkan bahwa semua Reservasi Kapasitas di semua kolam telah digunakan.

```
{ "CapacityReservationId":  "cr-111",
     "InstanceType":  "c5.xlarge",
     "AvailableInstanceCount":  0
}

 { "CapacityReservationId":  "cr-222",
     "InstanceType":  "c5.xlarge",
     "AvailableInstanceCount":  0
}
```

## Tutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Blok Kapasitas

Tutorial ini memandu Anda melalui langkah-langkah yang harus Anda lakukan sehingga Armada EC2 Anda meluncurkan instans ke Blok Kapasitas.

Dalam kebanyakan kasus, kapasitas target permintaan Armada EC2 harus kurang dari atau sama dengan kapasitas yang tersedia dari reservasi Blok Kapasitas yang Anda targetkan. Permintaan kapasitas target yang melebihi batas reservasi Blok Kapasitas tidak akan dipenuhi. Jika permintaan kapasitas target melebihi batas reservasi Blok Kapasitas Anda, Anda akan menerima nilai `Insufficient Capacity Exception` untuk kapasitas yang melebihi batas reservasi Blok Kapasitas Anda.

> **Note**
>
> Untuk Blok Kapasitas, Armada EC2 tidak akan mundur untuk meluncurkan On-Demand Instans untuk sisa kapasitas target yang diinginkan.

Jika Armada EC2 tidak dapat memenuhi kapasitas target yang diminta dalam reservasi Blok Kapasitas yang tersedia, Armada EC2 akan memenuhi kapasitas sebanyak mungkin dan mengembalikan instans yang dapat diluncurkan. Anda dapat mengulangi panggilan ke Armada EC2 lagi hingga semua instans disediakan.

Setelah mengonfigurasi permintaan Armada EC2, Anda harus menunggu hingga tanggal mulai reservasi Blok Kapasitas Anda. Jika Anda membuat permintaan ke Armada EC2 untuk meluncurkan ke Blok Kapasitas yang belum dimulai, Anda akan menerima. `Insufficient Capacity Error`

Setelah reservasi Blok Kapasitas aktif, Anda dapat membuat panggilan API Armada EC2 dan menyediakan instans ke dalam Blok Kapasitas berdasarkan parameter yang Anda pilih. Instans yang berjalan di Blok Kapasitas terus berjalan sampai Anda menghentikan atau menghentikannya secara manual atau hingga Amazon EC2 mengakhiri instans saat reservasi Blok Kapasitas berakhir.

Untuk informasi selengkapnya tentang Blok Kapasitas, lihatBlok Kapasitas untuk ML.

Pertimbangan-pertimbangan

- Hanya permintaan jenis EC2 Fleet yang `instant` didukung untuk meluncurkan instans ke Blok Kapasitas. Untuk informasi selengkapnya, lihat Konfigurasikan jenis Armada EC2 instant.
- Beberapa Blok Kapasitas dalam permintaan Armada EC2 yang sama tidak didukung.
- Menggunakan `OnDemandTargetCapacity` atau `SpotTargetCapacity` sekaligus juga mengatur `capacity-block` sebagai `DefaultTargetCapacity` tidak didukung.
- Jika `DefaultTargetCapacityType` diatur ke `capacity-block`, Anda tidak dapat menyediakan `OnDemandOptions::CapacityReservationOptions`. Pengecualian akan terjadi.

Untuk mengonfigurasi Armada EC2 untuk meluncurkan instans ke Blok Kapasitas

1. Buat template peluncuran.

   Dalam template peluncuran, lakukan hal berikut:

- Untuk`InstanceMarketOptionsRequest`, atur `MarketType` ke`capacity-block`.
- Untuk menargetkan reservasi Blok Kapasitas, for`CapacityReservationID`, tentukan ID reservasi Blok Kapasitas.

Catat nama dan versi template peluncuran. Anda akan menggunakan informasi ini pada langkah berikutnya.

Untuk informasi selengkapnya tentang membuat template peluncuran, lihatBuat template peluncuran Amazon EC2.

2. Konfigurasikan Armada EC2.

Buat file,`config.json`, dengan konfigurasi berikut untuk Armada EC2 Anda. Dalam contoh berikut, ganti pengidentifikasi sumber daya dengan pengidentifikasi sumber daya Anda sendiri.

Untuk informasi selengkapnya tentang mengonfigurasi Armada EC2, lihatMembuat Armada EC2.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "CBR-launch-template",
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "p5.48xlarge",
                    "AvailabilityZone": "us-east-1a"
                },
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 10,
        "DefaultTargetCapacityType": "capacity-block"
    },
    "Type": "instant"
}
```

3. Luncurkan armada.

Gunakan perintah create-fleet berikut.

```
aws ec2 create-fleet --cli-input-json file://config.json
```

Untuk informasi selengkapnya, lihat Membuat Armada EC2.

# Tutorial: Konfigurasikan Armada EC2 Anda untuk meluncurkan instans ke Reservasi Kapasitas yang Dapat Diinterupsi

Tutorial ini memandu Anda melalui langkah-langkah yang harus Anda lakukan agar Armada EC2 meluncurkan instans ke Reservasi Kapasitas Terputus.

Reservasi Kapasitas Terputus mewakili kapasitas cadangan yang dipinjamkan kepada Anda oleh pemilik Reservasi On-Demand Kapasitas dalam AWS organisasi Anda. Reservasi ini cocok untuk beban kerja yang dapat diinterupsi karena pemilik kapasitas dapat merebut kembali kapasitas kapan saja dengan pemberitahuan interupsi 2 menit, setelah itu EC2 mengakhiri instans. Untuk informasi selengkapnya tentang Reservasi Kapasitas Terputus, lihatReservasi Kapasitas yang Dapat Diinterupsi.

Kapasitas target permintaan Armada EC2 harus kurang dari atau sama dengan kapasitas yang tersedia dari Reservasi Kapasitas Terputus yang Anda targetkan. Jika permintaan kapasitas target melebihi kapasitas yang tersedia dari Reservasi Kapasitas Terputus Anda, EC2 Fleet meluncurkan instans sebanyak mungkin dan melaporkan instans yang diluncurkan dalam respons API.

Instans yang berjalan di Reservasi Kapasitas Terputus terus berjalan hingga Anda menghentikan atau menghentikannya secara manual, atau hingga pemilik kapasitas mengklaim kembali kapasitas. Saat pemilik kapasitas mengklaim kembali kapasitas, Amazon EC2 mengirimkan EventBridge pemberitahuan Amazon 2 menit sebelum mengakhiri instans.

## Pertimbangan-pertimbangan

- Hanya permintaan jenis EC2 Fleet yang `instant` didukung untuk meluncurkan instans ke Reservasi Kapasitas Terputus.
- Menggunakan `OnDemandTargetCapacity` atau `SpotTargetCapacity` sementara juga meny `reserved-capacity` etel `DefaultTargetCapacityType` karena tidak didukung.
- Saat Anda menentukan beberapa templat peluncuran, masing-masing menargetkan Reservasi Kapasitas Terputus yang berbeda, EC2 Fleet menyediakan instans di semua reservasi yang cocok.

- Untuk Reservasi Kapasitas Terputus, Armada EC2 tidak kembali meluncurkan On-Demand Instans atau Instans Spot untuk sisa kapasitas target yang diinginkan.

## Memverifikasi izin

Sebelum membuat Armada EC2, verifikasi bahwa Anda memiliki peran IAM dengan izin yang diperlukan. Untuk informasi selengkapnya, lihat Prasyarat Armada EC2.

Untuk meluncurkan instans ke Reservasi Kapasitas Terputus, Anda harus melakukan langkah-langkah berikut:

## Langkah 1: Buat templat peluncuran

Gunakan perintah create-launch-template untuk membuat template peluncuran yang menentukan Reservasi Kapasitas Terputus untuk ditargetkan. Di template peluncuran, atur `MarketType` ke `interruptible-capacity-reservation` dan tentukan Reserv `CapacityReservationId` asi Kapasitas Terputus Anda.

Contoh konfigurasi template peluncuran:

```
{
    "LaunchTemplateName": "interruptible-cr-launch-template",
    "LaunchTemplateData": {
        "InstanceType": "m5.large",
        "ImageId": "ami-0abcdef1234567890",
        "CapacityReservationSpecification": {
            "CapacityReservationTarget": {
                "CapacityReservationId": "cr-0123456789abcdef0"
            }
        },
        "InstanceMarketOptions": {
            "MarketType": "interruptible-capacity-reservation"
        }
    }
}
```

### Contoh Output

```
{
    "LaunchTemplate": {
        "LaunchTemplateId": "lt-0123456789example",
```

```
        "LaunchTemplateName": "interruptible-cr-launch-template",
        "CreateTime": "2026-03-12T10:00:00.000Z",
        "CreatedBy": "arn:aws:iam::123456789012:user/Admin",
        "DefaultVersionNumber": 1,
        "LatestVersionNumber": 1
    }
}
```

Untuk informasi selengkapnya, lihat Buat template peluncuran Amazon EC2.

## Langkah 2: Konfigurasikan Armada EC2

Buat file konfigurasi untuk Armada EC2 yang menentukan template peluncuran dan kapasitas target. Konfigurasi berikut menggunakan template peluncuran `interruptible-cr-launch-template` yang Anda buat di Langkah 1.

Anda harus menentukan `ReservedCapacityOptions` dengan `ReservationType` set ke `interruptible-capacity-reservation` saat menggunakan `reserved-capacity` sebagai`DefaultTargetCapacityType`.

Buat file bernama `config.json` dengan konten berikut:

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "interruptible-cr-launch-template",
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "m5.large",
                    "AvailabilityZone": "us-east-1a"
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 10,
        "DefaultTargetCapacityType": "reserved-capacity"
    },
    "ReservedCapacityOptions": {
        "ReservationType": ["interruptible-capacity-reservation"]
```

```
    },
    "Type": "instant"
}
```

Parameter konfigurasi utama:

| Parameter | Deskripsi |
| --- | --- |
| `DefaultTargetCapacityType` | Setel `reserved-capacity` untuk menunjukkan bahwa instance harus diluncurkan ke kapasitas cadangan. |
| `ReservedCapacityOptions` | Menentukan jenis kapasitas yang dicadangkan. Untuk Reservasi Kapasitas yang Dapat Diinterupsi, setel `ReservationType` ke`["interruptible-capacity-reservation"]` . |
| `Type` | Harus diatur ke `instant`. Hanya armada instan yang didukung untuk Reservasi Kapasitas Terputus. |

## Langkah 3: Luncurkan armada dan lihat hasilnya

Gunakan perintah create-fleet untuk membuat armada:

```
aws ec2 create-fleet \
    --cli-input-json file://config.json
```

Setelah Anda membuat `instant` armada menggunakan konfigurasi sebelumnya, EC2 Fleet meluncurkan 10 instans ke Reservasi Kapasitas Terputus untuk memenuhi kapasitas target.

> **Note**
>
> Jika armada tidak dapat memenuhi kapasitas target penuh, respons mencakup instans yang diluncurkan dan kesalahan apa pun untuk kapasitas yang tidak terpenuhi.

Contoh Output

```
{
    "FleetId": "fleet-12345678-1234-1234-1234-123456789012",
```

```
    "Instances": [
        {
            "LaunchTemplateAndOverrides": {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateId": "lt-0123456789example",
                    "Version": "1"
                },
                "Overrides": {
                    "InstanceType": "m5.large",
                    "AvailabilityZone": "us-east-1a"
                }
            },
            "Lifecycle": "interruptible-capacity-reservation",
            "InstanceIds": [
                "i-0123456789example1",
                "i-0123456789example2",
                "i-0123456789example3",
                "i-0123456789example4",
                "i-0123456789example5",
                "i-0123456789example6",
                "i-0123456789example7",
                "i-0123456789example8",
                "i-0123456789example9",
                "i-0123456789example0"
            ],
            "InstanceType": "m5.large",
            "Platform": "Linux/UNIX"
        }
    ],
    "Errors": []
}
```

Untuk informasi selengkapnya, lihat Membuat Armada EC2.

## Bersihkan

Untuk menghentikan tagihan, hentikan instans ketika tidak lagi diperlukan. Perhatikan bahwa EC2 juga mengakhiri instans yang diluncurkan ke Reservasi Kapasitas Terputus secara otomatis saat pemilik kapasitas mengklaim kembali kapasitas.

## Sumber daya terkait

- Reservasi Kapasitas yang Dapat Diinterupsi

- Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand
- Bekerja dengan EC2 Fleet
- Membuat Armada EC2
- Simpan parameter peluncuran instans di templat peluncuran Amazon EC2

# Tutorial: Mengkonfigurasi Armada EC2 Anda untuk meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas

Tutorial ini memandu Anda melalui langkah-langkah untuk mengonfigurasi Armada EC2 yang meluncurkan instans di beberapa jenis Reservasi Kapasitas — Reservasi On-Demand Kapasitas (ODCR), Blok Kapasitas untuk ML, dan Reservasi Kapasitas yang dapat diinterupsi — menggunakan Grup Sumber Daya Reservasi Kapasitas tunggal.

Anda akan mempelajari cara membuat Grup Sumber Daya Reservasi Kapasitas yang berisi jenis Reservasi Kapasitas yang berbeda, mengonfigurasi Armada EC2 Anda untuk memprioritaskan jenis Reservasi Kapasitas dalam urutan tertentu, dan secara opsional kembali ke On-Demand kapasitas ketika kapasitas yang dipesan tidak mencukupi untuk memenuhi target Anda.

## Pertimbangan-pertimbangan

- Hanya permintaan jenis Armada EC2 yang `instant` didukung saat menggunakan`ReservedCapacityOptions`.
- Anda harus secara eksplisit menentukan jenis Reservasi Kapasitas yang akan ditargetkan dalam `ReservationTypes` daftar. EC2 Fleet meluncurkan instans hanya ke jenis Reservasi Kapasitas yang Anda tentukan.
- `ReservedCapacityOptions`saling eksklusif dengan`OnDemandOptions.CapacityReservationOptions`. Anda tidak dapat menggunakan keduanya dalam permintaan Armada EC2 yang sama.

Tutorial ini mengasumsikan bahwa Anda sudah memiliki Reservasi Kapasitas berikut di akun Anda:

- ODCR (`cr-1234567890abcdef1`) untuk masuk `p5.48xlarge` `us-east-1a`
- ODCR (`cr-abcdef1234567890a`) untuk masuk `p4d.48xlarge` `us-east-1b`
- Blok Kapasitas (`cr-0123456789abcdef0`) untuk `p5.48xlarge` masuk `us-east-1a`

- Reservasi Kapasitas (`cr-9876543210fedcba9`) yang dapat diinterup `p5.48xlarge` si untuk `us-east-1a`, dibagikan dengan Anda oleh akun lain di organisasi Anda

## Memverifikasi izin

Sebelum membuat Armada EC2, verifikasi bahwa Anda memiliki peran IAM dengan izin yang diperlukan. Untuk informasi selengkapnya, lihat Prasyarat Armada EC2.

## Langkah 1: Buat Grup Sumber Daya Reservasi Kapasitas

Buat Grup Sumber Daya Reservasi Kapasitas untuk menyimpan Reservasi Kapasitas yang ditargetkan Armada EC2 Anda. Konfigurasikan grup sehingga menerima Reservasi Kapasitas dari jenis instans dan jenis Reservasi Kapasitas apa pun, yang memungkinkan grup menyimpan campuran jenis Reservasi Kapasitas. Untuk petunjuk langkah demi langkah, lihat Grup Reservasi Kapasitas.

> **Note**
>
> Saat Anda membuat grup, jangan tentukan `reservation-type` parameter `instance-type` atau dalam `AWS::EC2::CapacityReservationPool` konfigurasi. Menghilangkan parameter ini memungkinkan grup menerima Reservasi Kapasitas dari jenis instans dan jenis Reservasi Kapasitas apa pun.

## Langkah 2: Tambahkan Reservasi Kapasitas ke grup

Tambahkan Reservasi Kapasitas Anda ke Grup Sumber Daya Reservasi Kapasitas. Anda dapat menambahkan kombinasi ODCR, Blok Kapasitas untuk ML, Reservasi Kapasitas yang dapat diinterupsi, dan UltraServer Blok Kapasitas. Untuk tutorial ini, tambahkan empat Reservasi Kapasitas yang tercantum di awal tutorial ini. Untuk petunjuk langkah demi langkah, lihat Grup Reservasi Kapasitas.

## Langkah 3: Buat template peluncuran

Gunakan perintah create-launch-template untuk membuat template peluncuran yang menargetkan Grup Sumber Daya Reservasi Kapasitas. Saat Anda menggunakan konfigurasi `ReservedCapacityOptions` Armada EC2 Anda, Anda tidak perlu menyetel template peluncuran — `MarketType` Armada EC2 menyetel ini secara otomatis untuk setiap jenis Reservasi Kapasitas.

```
aws ec2 create-launch-template \
    --launch-template-name my-cr-resource-group-template \
    --launch-template-data \
        '{"ImageId": "ami-0123456789example",
          "CapacityReservationSpecification":
            {"CapacityReservationTarget":
                {"CapacityReservationResourceGroupArn": "arn:aws:resource-groups:us-
east-1:123456789012:group/my-cr-resource-group"}
            }
        }'
```

## Langkah 4: Buat Armada EC2

Buat Armada EC2 yang menargetkan beberapa jenis Reservasi Kapasitas di Grup Sumber Daya Reservasi Kapasitas Anda. Konfigurasi kuncinya adalah`ReservedCapacityOptions`, yang menentukan yang berikut:

- `ReservationTypes`— Daftar tipe Reservasi Kapasitas yang diurutkan untuk ditargetkan. Armada EC2 bekerja melalui tipe dalam urutan ini. Nilai yang valid adalah `on-demand-capacity-reservation`, `capacity-block`, dan `interruptible-capacity-reservation`.
- `AllocationStrategy`— Bagaimana jenis instans diprioritaskan dalam setiap jenis Reservasi Kapasitas. Setel `prioritized` untuk mengikuti urutan penggantian jenis instans Anda.
- `ReservedCapacityFallbackOptions`— Setel `MarketTypes ["on-demand"]` untuk meluncurkan On-Demand Instans ketika kapasitas cadangan tidak mencukupi. Jika Anda tidak menentukan opsi fallback, Armada EC2 tidak akan kembali ke jenis pasar lainnya setelah jenis Reservasi Kapasitas yang ditentukan habis.

Buat file dengan nama `config.json` dengan konten berikut ini. Dalam contoh berikut, ganti pengidentifikasi sumber daya dengan pengidentifikasi sumber daya Anda sendiri.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-cr-resource-group-template",
                "Version": "1"
            },
            "Overrides": [
                {
```

```
                    "InstanceType": "p5.48xlarge",
                    "AvailabilityZone": "us-east-1a",
                    "Priority": 1
                },
                {
                    "InstanceType": "p4d.48xlarge",
                    "AvailabilityZone": "us-east-1b",
                    "Priority": 2
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 100,
        "DefaultTargetCapacityType": "reserved-capacity"
    },
    "ReservedCapacityOptions": {
        "ReservationTypes": ["on-demand-capacity-reservation", "capacity-block",
 "interruptible-capacity-reservation"],
        "ReservedCapacityFallbackOptions": {
            "MarketTypes": ["on-demand"]
        },
        "AllocationStrategy": "prioritized"
    },
    "Type": "instant"
}
```

Gunakan perintah create-fleet untuk meluncurkan armada:

```
aws ec2 create-fleet --cli-input-json file://config.json
```

Untuk informasi selengkapnya, lihat Membuat Armada EC2.

Bagaimana Armada EC2 menyelesaikan Reservasi Kapasitas

Berdasarkan konfigurasi sebelumnya, EC2 Fleet mencoba meluncurkan instans dalam urutan berikut:

1. ODCR — `p5.48xlarge` reservasi terlebih dahulu (jenis instans prioritas tertinggi), dan kemudian `p4d.48xlarge` reservasi. Dalam setiap jenis instance, jika ada beberapa ODCR, satu dipilih secara acak.

2. Blok Kapasitas — Jika tidak ada kapasitas yang tersedia di semua ODCR, Armada EC2 menargetkan Blok Kapasitas. `p5.48xlarge`reservasi terlebih dahulu, dan kemudian`p4d.48xlarge`, dipilih secara acak dalam setiap jenis instance.
3. Reservasi Kapasitas yang Dapat Diinterup si — Jika tidak ada kapasitas yang tersedia di semua Blok Kapasitas, Armada EC2 menargetkan Reservasi Kapasitas yang dapat diinterupsi. `p5.48xlarge`reservasi terlebih dahulu, dan kemudian`p4d.48xlarge`, dipilih secara acak dalam setiap jenis instance.
4. On-Demand fallback — Jika kapasitas cadangan tidak mencukupi untuk memenuhi target 100 instans, kapasitas yang tersisa diluncurkan sebagai On-Demand Instans. EC2 Fleet meluncurkan On-Demand Instans ini sesuai dengan strategi alokasi yang Anda tentukan. `OnDemandOptions`

> **Note**
>
> EC2 Fleet menyelesaikan jenis Reservasi Kapasitas terlebih dahulu, lalu jenis instans dalam setiap jenis Reservasi Kapasitas. Setiap jenis Reservasi Kapasitas habis sepenuhnya sebelum Armada EC2 pindah ke tipe berikutnya dalam daftar yang diurutkan.

## (Opsional) Langkah 5: Verifikasi properti instance

Gunakan perintah describe-instance untuk memverifikasi jenis Reservasi Kapasitas yang digunakan setiap instans. Instans yang diluncurkan oleh armada menampilkan properti berikut.

| Properti | On-Demand Nilai reservasi kapasitas | Nilai Blok Kapasitas | Nilai Reservasi Kapasitas yang Dapat Diinterupsi | On-Demand nilai |
|---|---|---|---|---|
| `instance-lifecycle` | `null` | `capacity-block` | `interruptible-capacity-reservation` | `null` |
| `capacity-reservation-id` | ID Reservasi Kapasitas | ID Reservasi Kapasitas | ID Reservasi Kapasitas yang dikonsumsi | Tidak hadir |

| Properti | On-Demand Nilai reservasi kapasitas yang dikonsumsi | Nilai Blok Kapasitas yang dikonsumsi | Nilai Reservasi Kapasitas yang Dapat Diinterupsi | On-Demand nilai |
|---|---|---|---|---|
| `capacity-reservation-specification` | Grup ARN | Grup ARN | Grup ARN | Tidak hadir |

Anda dapat memfilter instance berdasarkan Grup Sumber Daya Reservasi Kapasitas menggunakan perintah berikut:

```
aws ec2 describe-instances \
    --filters "Name=capacity-reservation-specification.capacity-reservation-target.capacity-reservation-resource-group-arn,Values=arn:aws:resource-groups:us-east-1:123456789012:group/my-cr-resource-group"
```

## Bersihkan

Untuk menghentikan tagihan, hentikan instans ketika tidak lagi diperlukan. Perhatikan bahwa instans yang diluncurkan ke Blok Kapasitas secara otomatis dihentikan saat reservasi Blok Kapasitas berakhir. Instans yang diluncurkan ke Reservasi Kapasitas yang dapat diinterupsi secara otomatis dihentikan jika pemilik kapasitas mengklaim kembali kapasitas tersebut.

## Sumber daya terkait

- Grup Reservasi Kapasitas
- Blok Kapasitas untuk ML
- Reservasi Kapasitas yang Dapat Diinterupsi
- Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand
- Bekerja dengan EC2 Fleet
- Membuat Armada EC2
- Simpan parameter peluncuran instans di templat peluncuran Amazon EC2

# Contoh konfigurasi CLI untuk Armada EC2

Anda dapat menentukan konfigurasi Armada EC2 Anda dalam file JSON, dan kemudian mereferensikan file tersebut dengan perintah create-fleet untuk membuat armada Anda, sebagai berikut:

```
aws ec2 create-fleet --cli-input-json file://file_name.json
```

Contoh berikut menggambarkan konfigurasi peluncuran untuk berbagai kasus penggunaan Armada EC2. Untuk informasi selengkapnya tentang parameter konfigurasi, lihat create-fleet.

Contoh

- Contoh 1: Meluncurkan Instans Spot sebagai opsi pembelian default
- Contoh 2: Luncur On-Demand kan Instans sebagai opsi pembelian default
- Contoh 3: Luncur On-Demand kan Instans sebagai kapasitas utama
- Contoh 4: Luncurkan On-Demand Instans menggunakan beberapa Reservasi Kapasitas
- Contoh 5: Luncur On-Demand kan Instans menggunakan Reservasi Kapasitas ketika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak digunakan
- Contoh 6: Meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan
- Contoh 7: Konfigurasikan Penyeimbangan Kembali Kapasitas untuk meluncurkan Instans Spot pengganti
- Contoh 8: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas
- Contoh 9: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas
- Contoh 10: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas harga
- Contoh 11: Mengkonfigurasi pemilihan jenis instans berbasis atribut
- Contoh 12: Meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas

Untuk contoh CLI lainnya untuk jenis armada`instant`, lihat. Konfigurasikan jenis Armada EC2 instant

## Contoh 1: Meluncurkan Instans Spot sebagai opsi pembelian default

Contoh berikut menentukan parameter minimum yang diperlukan dalam Armada EC2: templat peluncuran, kapasitas target, dan opsi pembelian default. Templat peluncuran diidentifikasi dengan

ID templat dan nomor versi peluncurannya. Kapasitas target untuk armada adalah 2 instans, dan opsi pembelian default adalah `spot`, yang menghasilkan armada meluncurkan 2 Instans Spot.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "DefaultTargetCapacityType": "spot"
    }
}
```

## Contoh 2: Luncur On-Demand kan Instans sebagai opsi pembelian default

Contoh berikut menentukan parameter minimum yang diperlukan dalam Armada EC2: templat peluncuran, kapasitas target, dan opsi pembelian default. Templat peluncuran diidentifikasi dengan ID templat dan nomor versi peluncurannya. Kapasitas target untuk armada adalah 2 instance, dan opsi pembelian default adalahon-demand, yang menghasilkan armada meluncurkan 2 On-Demand Instans.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "DefaultTargetCapacityType": "on-demand"
    }
```

```
}
```

## Contoh 3: Luncur On-Demand kan Instans sebagai kapasitas utama

Contoh berikut menentukan kapasitas target total 2 instans untuk armada, dan kapasitas target 1 On-Demand Instans. Opsi pembelian default adalah `spot`. Armada meluncurkan 1 On-Demand Instans seperti yang ditentukan, tetapi perlu meluncurkan satu instance lagi untuk memenuhi kapasitas target total. Opsi pembelian untuk selisihnya dihitung sebagai `TotalTargetCapacity` – `OnDemandTargetCapacity` = `DefaultTargetCapacityType`, yang menghasilkan armada yang meluncurkan 1 Instans Spot.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            }

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 2,
        "OnDemandTargetCapacity": 1,
        "DefaultTargetCapacityType": "spot"
    }
}
```

## Contoh 4: Luncurkan On-Demand Instans menggunakan beberapa Reservasi Kapasitas

Anda dapat mengonfigurasi armada untuk menggunakan Reservasi On-Demand Kapasitas terlebih dahulu saat meluncurkan On-Demand Instans dengan menyetel strategi penggunaan untuk Reservasi Kapasitas ke`use-capacity-reservations-first`. Contoh ini menunjukkan cara armada memilih Reservasi Kapasitas yang akan digunakan jika terdapat lebih banyak Reservasi Kapasitas daripada yang dibutuhkan untuk memenuhi kapasitas target.

Dalam contoh ini, konfigurasi armada adalah sebagai berikut:

- Kapasitas target: 12 On-Demand Instans

- Total Reservasi Kapasitas yang tidak digunakan: 15 (lebih dari kapasitas target armada 12 On-Demand Instans)
- Jumlah kolam Reservasi Kapasitas: 3 (`m5.large`, `m4.xlarge`, dan `m4.2xlarge`)
- Jumlah Reservasi Kapasitas per kolam: 5
- On-Demand strategi alokasi: `lowest-price` (Jika ada beberapa Reservasi Kapasitas yang tidak digunakan di beberapa kumpulan instans, armada menentukan kumpulan untuk meluncurkan On-Demand Instans berdasarkan strategi On-Demand alokasi.)

  Perhatikan bahwa Anda juga dapat menggunakan strategi alokasi `prioritized` alih-alih strategi alokasi `lowest-price`.

Reservasi Kapasitas

Akun tersebut memiliki 15 Reservasi Kapasitas yang tidak terpakai dalam 3 kolam yang berbeda. Jumlah Reservasi Kapasitas di setiap kolam ditunjukkan dengan `AvailableInstanceCount`.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
```

```
    "AvailableInstanceCount":5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

Konfigurasi Armada

Konfigurasi armada berikut hanya menampilkan konfigurasi terkait untuk contoh ini. Total kapasitas target adalah 12, dan tipe kapasitas target default adalah `on-demand`. Strateg On-Demand i alokasi adalah `lowest-price` Strategi penggunaan untuk Reservasi Kapasitas adalah `use-capacity-reservations-first`.

Dalam contoh ini, harga On-Demand Instance adalah:

- `m5.large` – 0,096 USD per jam
- `m4.xlarge` – 0,20 USD per jam
- `m4.2xlarge` – 0,40 USD per jam

> **Note**
>
> Tipe armada harus bertipe `instant`. Tipe armada lainnya tidak mendukung `use-capacity-reservations-first`.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-abc1234567example",
                "Version": "1"
            },
            "Overrides": [
                {
                  "InstanceType": "m5.large",
                  "AvailabilityZone": "us-east-1a",
                  "WeightedCapacity": 1
                },
                {
                  "InstanceType": "m4.xlarge",
                  "AvailabilityZone": "us-east-1a",
```

```
                "WeightedCapacity": 1
              },
              {
                "InstanceType": "m4.2xlarge",
                "AvailabilityZone": "us-east-1a",
                "WeightedCapacity": 1
              }
            ]

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 12,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions": {
            "UsageStrategy": "use-capacity-reservations-first"
        }
    },
    "Type": "instant"
}
```

Setelah Anda membuat armada `instant` menggunakan konfigurasi sebelumnya, 12 instans berikut diluncurkan untuk memenuhi kapasitas target:

- 5 `m5.large` On-Demand Instans in `us-east-1a` — `m5.large` in `us-east-1a` adalah harga terendah, dan ada 5 Reservasi `m5.large` Kapasitas yang tidak digunakan yang tersedia
- 5 `m4.xlarge` On-Demand Instans di us-east-1a — `m4.xlarge` in `us-east-1a` adalah harga terendah berikutnya, dan ada 5 Reservasi Kapasitas yang tidak digunakan yang tersedia `m4.xlarge`
- 2 `m4.2xlarge` On-Demand Instans di us-east-1a — `m4.2xlarge` in `us-east-1a` adalah harga terendah ketiga, dan ada 5 Reservasi `m4.2xlarge` Kapasitas yang tidak digunakan yang tersedia di mana hanya 2 yang diperlukan untuk memenuhi kapasitas target

Setelah armada diluncurkan, Anda dapat menjalankan describe-capacity-reservations untuk melihat jumlah Reservasi Kapasitas yang tidak terpakai yang tersisa. Dalam contoh ini, Anda akan melihat respons berikut, yang menunjukkan bahwa semua Reservasi Kapasitas `m5.large` dan `m4.xlarge` digunakan, dengan 3 Reservasi Kapasitas `m4.2xlarge` yang masih belum digunakan.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "AvailableInstanceCount": 3
}
```

## Contoh 5: Luncur On-Demand kan Instans menggunakan Reservasi Kapasitas ketika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak digunakan

Anda dapat mengonfigurasi armada untuk menggunakan Reservasi On-Demand Kapasitas terlebih dahulu saat meluncurkan On-Demand Instans dengan menyetel strategi penggunaan untuk Reservasi Kapasitas ke`use-capacity-reservations-first`. Contoh ini menunjukkan bagaimana armada memilih kumpulan instans untuk meluncurkan On-Demand Instans ketika total kapasitas target melebihi jumlah Reservasi Kapasitas yang tidak terpakai yang tersedia.

Dalam contoh ini, konfigurasi armada adalah sebagai berikut:

- Kapasitas target: 16 On-Demand Instans
- Total Reservasi Kapasitas yang tidak digunakan: 15 (kurang dari kapasitas target armada 16 On-Demand Instans)
- Jumlah kolam Reservasi Kapasitas: 3 (`m5.large`, `m4.xlarge`, dan `m4.2xlarge`)
- Jumlah Reservasi Kapasitas per kolam: 5
- On-Demand strategi alokasi: `lowest-price` (Jika jumlah Reservasi Kapasitas yang tidak digunakan kurang dari kapasitas On-Demand target, armada menentukan kumpulan tempat untuk meluncurkan On-Demand kapasitas yang tersisa berdasarkan strategi al On-Demand okasi.)

Perhatikan bahwa Anda juga dapat menggunakan strategi alokasi `prioritized` alih-alih strategi alokasi `lowest-price`.

Reservasi Kapasitas

Akun tersebut memiliki 15 Reservasi Kapasitas yang tidak terpakai dalam 3 kolam yang berbeda. Jumlah Reservasi Kapasitas di setiap kolam ditunjukkan dengan `AvailableInstanceCount`.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "m5.large",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount":5,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

Konfigurasi Armada

Konfigurasi armada berikut hanya menampilkan konfigurasi terkait untuk contoh ini. Total kapasitas target adalah 16, dan tipe kapasitas target default adalah `on-demand`. Strateg On-Demand i alokasi adalah `lowest-price` Strategi penggunaan untuk Reservasi Kapasitas adalah `use-capacity-reservations-first`.

Dalam contoh ini, harga On-Demand Instance adalah:

- m5.large – 0,096 USD per jam
- m4.xlarge – 0,20 USD per jam
- m4.2xlarge – 0,40 USD per jam

> **Note**
>
> Tipe armada harus `instant`. Tipe armada lainnya tidak mendukung `use-capacity-reservations-first`.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateId": "lt-0e8c754449b27161c",
                "Version": "1"
            },
            "Overrides": [
                {
                  "InstanceType": "m5.large",
                  "AvailabilityZone": "us-east-1a",
                  "WeightedCapacity": 1
                },
                {
                  "InstanceType": "m4.xlarge",
                  "AvailabilityZone": "us-east-1a",
                  "WeightedCapacity": 1
                },
                {
                  "InstanceType": "m4.2xlarge",
                  "AvailabilityZone": "us-east-1a",
                  "WeightedCapacity": 1
                }
```

```
            ]

        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 16,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price"
        "CapacityReservationOptions": {
            "UsageStrategy": "use-capacity-reservations-first"
        }
    },
    "Type": "instant",
}
```

Setelah Anda membuat armada `instant` menggunakan konfigurasi sebelumnya, 16 instans berikut diluncurkan untuk memenuhi kapasitas target:

- 6 `m5.large` On-Demand Instans in `us-east-1a` — `m5.large` in `us-east-1a` adalah harga terendah, dan ada 5 Reservasi `m5.large` Kapasitas yang tidak digunakan yang tersedia. Reservasi Kapasitas digunakan terlebih dahulu untuk meluncurkan 5 On-Demand Instans. Setelah sisa `m4.xlarge` dan Reservasi `m4.2xlarge` Kapasitas digunakan, untuk memenuhi kapasitas target, On-Demand Instans tambahan diluncurkan sesuai dengan strategi al On-Demand okasi, yang ada `lowest-price` dalam contoh ini.
- 5 `m4.xlarge` On-Demand Instans in `us-east-1a` — `m4.xlarge` in `us-east-1a` adalah harga terendah berikutnya, dan ada 5 Reservasi `m4.xlarge` Kapasitas yang tidak terpakai yang tersedia
- 5 `m4.2xlarge` On-Demand Instans in `us-east-1a` — `m4.2xlarge` in `us-east-1a` adalah harga terendah ketiga, dan ada 5 Reservasi `m4.2xlarge` Kapasitas yang tidak terpakai yang tersedia

Setelah armada diluncurkan, Anda dapat menjalankan describe-capacity-reservations untuk melihat jumlah Reservasi Kapasitas yang tidak terpakai yang tersisa. Dalam contoh ini, Anda akan melihat respons berikut, yang menunjukkan bahwa semua Reservasi Kapasitas di semua kolam telah digunakan.

```
{
    "CapacityReservationId": "cr-111",
```

```
    "InstanceType": "m5.large",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "m4.xlarge",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-333",
    "InstanceType": "m4.2xlarge",
    "AvailableInstanceCount": 0
}
```

## Contoh 6: Meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan

Anda dapat mengonfigurasi armada untuk menggunakan Reservasi `targeted` On-Demand Kapasitas terlebih dahulu saat meluncurkan On-Demand Instans dengan menyetel strategi penggunaan untuk Reservasi Kapasitas ke`use-capacity-reservations-first`. Contoh ini menunjukkan cara meluncurkan On-Demand Instans ke `targeted` Reservasi Kapasitas, di mana atribut Reservasi Kapasitas sama kecuali untuk Zona Ketersediaan (`us-east-1a`dan`us-east-1b`). Ini juga menunjukkan bagaimana armada memilih kumpulan instans untuk meluncurkan On-Demand Instans ketika kapasitas target total melebihi jumlah Reservasi Kapasitas yang tidak terpakai yang tersedia.

Dalam contoh ini, konfigurasi armada adalah sebagai berikut:

- Kapasitas target: 10 On-Demand Instans
- Total Reserv `targeted` asi Kapasitas yang tidak digunakan: 6 (kurang dari kapasitas On-Demand target armada 10 On-Demand Instans)
- Jumlah kolam Reservasi Kapasitas: 2 (`us-east-1a` dan `us-east-1b`)
- Jumlah Reservasi Kapasitas per kolam: 3
- On-Demand strategi alokasi: `lowest-price` (Jika jumlah Reservasi Kapasitas yang tidak digunakan kurang dari kapasitas On-Demand target, armada menentukan kumpulan tempat untuk meluncurkan On-Demand kapasitas yang tersisa berdasarkan strategi al On-Demand okasi.)

Perhatikan bahwa Anda juga dapat menggunakan strategi alokasi `prioritized` alih-alih strategi alokasi `lowest-price`.

Untuk panduan prosedur yang harus Anda lakukan untuk menyelesaikan contoh ini, lihat Tutorial: Mengkonfigurasi Armada EC2 untuk meluncurkan On-Demand Instans menggunakan Reservasi Kapasitas yang ditargetkan.

Reservasi Kapasitas

Akun tersebut memiliki 6 Reservasi Kapasitas yang tidak terpakai dalam 2 kolam yang berbeda. Dalam contoh ini, kolam berbeda-beda menurut Zona Ketersediaannya. Jumlah Reservasi Kapasitas di setiap kolam ditunjukkan dengan `AvailableInstanceCount`.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "c5.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1a",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.xlarge",
    "InstancePlatform": "Linux/UNIX",
    "AvailabilityZone": "us-east-1b",
    "AvailableInstanceCount": 3,
    "InstanceMatchCriteria": "open",
    "State": "active"
}
```

Konfigurasi Armada

Konfigurasi armada berikut hanya menampilkan konfigurasi terkait untuk contoh ini. Total kapasitas target adalah 10, dan tipe kapasitas target default adalah `on-demand`. Strateg On-Demand i alokasi adalah `lowest-price` Strategi penggunaan untuk Reservasi Kapasitas adalah `use-capacity-reservations-first`.

Dalam contoh ini, harga On-Demand Instans untuk `c5.xlarge` in `us-east-1` adalah $0,17 per jam.

> **Note**
>
> Tipe armada harus `instant`. Tipe armada lainnya tidak mendukung `use-capacity-reservations-first`.

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
            "Overrides": [
               {
                   "InstanceType": "c5.xlarge",
                   "AvailabilityZone": "us-east-1a"
               },
               {
                    "InstanceType": "c5.xlarge",
                    "AvailabilityZone": "us-east-1b"
               }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 10,
        "DefaultTargetCapacityType": "on-demand"
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price",
        "CapacityReservationOptions": {
             "UsageStrategy": "use-capacity-reservations-first"
         }
    },
    "Type": "instant"
}
```

Setelah Anda membuat armada `instant` menggunakan konfigurasi sebelumnya, 10 instans berikut diluncurkan untuk memenuhi kapasitas target:

- Reservasi Kapasitas digunakan terlebih dahulu untuk meluncurkan 6 On-Demand Instans sebagai berikut:
  - 3 On-Demand Instans diluncurkan ke 3 Reservasi `c5.xlarge` `targeted` Kapasitas di `us-east-1a`
  - 3 On-Demand Instans diluncurkan ke 3 Reservasi `c5.xlarge` `targeted` Kapasitas di `us-east-1b`
- Untuk memenuhi kapasitas target, 4 On-Demand Instans tambahan diluncurkan ke On-Demand kapasitas reguler sesuai dengan strategi On-Demand alokasi, yang ada `lowest-price` dalam contoh ini. Namun, karena kumpulan memiliki harga yang sama (karena harga per Wilayah dan bukan per Zona Ketersediaan), armada meluncurkan 4 On-Demand Instans yang tersisa ke salah satu kumpulan.

Setelah armada diluncurkan, Anda dapat menjalankan describe-capacity-reservations untuk melihat jumlah Reservasi Kapasitas yang tidak terpakai yang tersisa. Dalam contoh ini, Anda akan melihat respons berikut, yang menunjukkan bahwa semua Reservasi Kapasitas di semua kolam telah digunakan.

```
{
    "CapacityReservationId": "cr-111",
    "InstanceType": "c5.xlarge",
    "AvailableInstanceCount": 0
}

{
    "CapacityReservationId": "cr-222",
    "InstanceType": "c5.xlarge",
    "AvailableInstanceCount": 0
}
```

## Contoh 7: Konfigurasikan Penyeimbangan Kembali Kapasitas untuk meluncurkan Instans Spot pengganti

Contoh berikut mengonfigurasi Armada EC2 untuk meluncurkan Instans Spot pengganti saat Amazon EC2 memancarkan rekomendasi penyeimbangan ulang untuk Instans Spot di armada. Untuk mengonfigurasi penggantian otomatis Instans Spot, untuk`ReplacementStrategy` , tentukan`launch-before-terminate` . Untuk mengonfigurasi penundaan waktu dari saat Instans Spot pengganti baru diluncurkan hingga saat Instans Spot lama dihapus secara otomatis, untuk

`termination-delay`, tentukan nilai dalam detik. Untuk informasi selengkapnya, lihat Opsi konfigurasi.

> **Note**
>
> Sebaiknya gunakan `launch-before-terminate` hanya jika Anda dapat memprediksi lamanya prosedur pematian instans akan selesai sehingga instans lama hanya dihentikan setelah prosedur ini selesai. Anda dikenai biaya untuk semua instans saat semuanya berjalan.

Efektivitas strategi Penyeimbangan Ulang Kapasitas bergantung pada jumlah kolam kapasitas Spot yang ditentukan dalam permintaan Armada EC2. Sebaiknya konfigurasikan armada dengan set tipe instans dan Zona Ketersediaan yang beragam, dan untuk `AllocationStrategy`, tentukan `capacity-optimized`. Untuk informasi selengkapnya tentang hal-hal yang harus Anda pertimbangkan saat mengonfigurasi Armada EC2 untuk Penyeimbangan Ulang Kapasitas, lihat Gunakan Penyeimbangan Kembali Kapasitas di Armada EC2 dan Spot Fleet untuk menggantikan Instans Spot yang berisiko.

```
{
    "ExcessCapacityTerminationPolicy": "termination",
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "LaunchTemplate",
                "Version": "1"
            },
                 "Overrides": [
                       {
                           "InstanceType": "c3.large",
                           "WeightedCapacity": 1,
                            "Placement": {
                               "AvailabilityZone": "us-east-1a"
                           }
                       },
                       {
                           "InstanceType": "c4.large",
                           "WeightedCapacity": 1,
                            "Placement": {
                               "AvailabilityZone": "us-east-1a"
                           }
```

```
                        },
                        {
                            "InstanceType": "c5.large",
                            "WeightedCapacity": 1,
                             "Placement": {
                                "AvailabilityZone": "us-east-1a"
                            }
                        }
                ]
          }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 5,
        "DefaultTargetCapacityType": "spot"
    },
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        "MaintenanceStrategies": {
            "CapacityRebalance": {
                "ReplacementStrategy": "launch-before-terminate",
                "TerminationDelay": "720"
            }
        }
    }
}
```

## Contoh 8: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas

Contoh berikut menunjukkan cara mengonfigurasi Armada EC2 dengan strategi alokasi yang mengoptimalkan kapasitas. Untuk mengoptimalkan kapasitas, Anda harus mengatur `AllocationStrategy` ke `capacity-optimized`.

Pada contoh berikut ini, tiga spesifikasi peluncuran menentukan tiga kolam kapasitas Spot. Kapasitas target adalah 50 Instans Spot. Armada EC2 berupaya meluncurkan 50 Instans Spot ke kolam kapasitas Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized",
        },
    "LaunchTemplateConfigs": [
```

```
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
                 "Overrides": [
                       {
                           "InstanceType": "r4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2a"
                           },
                      },
                       {
                           "InstanceType": "m4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           },
                       },
                       {
                           "InstanceType": "c5.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           }
                       }
                 ]
            }
    ],
    "TargetCapacitySpecification": {
            "TotalTargetCapacity": 50,
            "DefaultTargetCapacityType": "spot"

    }
}
```

## Contoh 9: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas

Contoh berikut menunjukkan cara mengonfigurasi Armada EC2 dengan strategi alokasi Spot yang mengoptimalkan kapasitas sambil menggunakan prioritas dengan upaya terbaik.

Jika menggunakan strategi alokasi `capacity-optimized-prioritized`, Anda dapat menggunakan parameter `Priority` untuk menentukan prioritas kolam kapasitas Spot, yaitu makin

rendah angkanya, makin tinggi prioritasnya. Anda juga dapat mengatur prioritas yang sama untuk beberapa kolam kapasitas Spot jika Anda menginginkannya setara. Jika Anda tidak menetapkan prioritas, kolam akan dianggap yang terakhir dalam hal prioritas.

Untuk memprioritaskan kolam kapasitas Spot, Anda harus mengatur `AllocationStrategy` ke `capacity-optimized-prioritized`. Armada EC2 akan mengoptimalkan kapasitas terlebih dahulu, tetapi akan mempertimbangkan prioritas dengan upaya terbaik (misalnya, jika mempertimbangkan prioritas tidak akan secara signifikan memengaruhi kemampuan Armada EC2 untuk menyediakan kapasitas optimal). Ini adalah pilihan opsi yang bagus untuk beban kerja di mana kemungkinan gangguan harus diminimalkan dan preferensi untuk tipe instans tertentu menjadi penting.

Pada contoh berikut ini, tiga spesifikasi peluncuran menentukan tiga kolam kapasitas Spot. Setiap kolam diprioritaskan, yaitu makin rendah jumlahnya, makin tinggi prioritasnya. Kapasitas target adalah 50 Instans Spot. Armada EC2 berupaya meluncurkan 50 Instans Spot ke dalam kolam kapasitas Spot dengan prioritas tertinggi menggunakan upaya terbaik, tetapi mengoptimalkan kapasitas terlebih dahulu.

```
{
    "SpotOptions": {
        "AllocationStrategy": "capacity-optimized-prioritized"
        },
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
                 "Overrides": [
                        {
                           "InstanceType": "r4.2xlarge",
                           "Priority": 1,
                           "Placement": {
                               "AvailabilityZone": "us-west-2a"
                           },
                      },
                       {
                           "InstanceType": "m4.2xlarge",
                           "Priority": 2,
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
```

```
                        },
                    },
                    {
                        "InstanceType": "c5.2xlarge",
                        "Priority": 3,
                        "Placement": {
                            "AvailabilityZone": "us-west-2b"
                        }
                    }
                ]
            }
    ],
    "TargetCapacitySpecification": {
            "TotalTargetCapacity": 50,
            "DefaultTargetCapacityType": "spot"
}
```

## Contoh 10: Luncurkan Instans Spot dalam armada yang dioptimalkan kapasitas harga

Contoh berikut menunjukkan cara mengonfigurasi Armada EC2 dengan strategi alokasi Spot yang mengoptimalkan kapasitas dan harga terendah. Untuk mengoptimalkan kapasitas sambil mempertimbangkan harga, Anda harus mengatur Spot `AllocationStrategy` ke `price-capacity-optimized`.

Pada contoh berikut ini, tiga spesifikasi peluncuran menentukan tiga kolam kapasitas Spot. Kapasitas target adalah 50 Instans Spot. Armada EC2 berupaya meluncurkan 50 Instans Spot ke kolam kapasitas Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan sekaligus memilih kolam yang memiliki harga terendah.

```
{
    "SpotOptions": {
        "AllocationStrategy": "price-capacity-optimized",
        "MinTargetCapacity": 2,
        "SingleInstanceType": true
    },
    "OnDemandOptions": {
        "AllocationStrategy": "lowest-price"
    },
    "LaunchTemplateConfigs": [
        {
```

```
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
                 "Overrides": [
                       {
                           "InstanceType": "r4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2a"
                           },
                      },
                       {
                           "InstanceType": "m4.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           },
                       },
                       {
                           "InstanceType": "c5.2xlarge",
                           "Placement": {
                               "AvailabilityZone": "us-west-2b"
                           }
                       }
                  ]
           }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 50,
        "OnDemandTargetCapacity":0,
        "SpotTargetCapacity":50,
        "DefaultTargetCapacityType": "spot"
    },
    "Type": "instant"
}
```

## Contoh 11: Mengkonfigurasi pemilihan jenis instans berbasis atribut

Contoh berikut menunjukkan cara mengonfigurasi Armada EC2 untuk menggunakan pemilihan tipe instans berbasis atribut untuk mengidentifikasi tipe instans. Untuk menentukan atribut instans yang diperlukan, Anda menentukan atribut dalam struktur `InstanceRequirements`.

Pada contoh berikut ini, dua atribut instans ditentukan:

- `VCpuCount` – Minimum 2 vCPU ditentukan. Karena tidak ada jumlah maksimum yang ditentukan, maka tidak ada batas maksimum.
- `MemoryMiB` – Minimum 4 MiB memori ditentukan. Karena tidak ada jumlah maksimum yang ditentukan, maka tidak ada batas maksimum.

Setiap tipe instans yang memiliki 2 atau lebih VCPU dan 4 MiB atau lebih memori akan diidentifikasi. Namun, perlindungan harga dan strategi alokasi mungkin akan mengecualikan beberapa tipe instans jika Armada EC2 menyediakan armada.

Untuk daftar dan deskripsi semua atribut yang mungkin dapat Anda tentukan, lihat InstanceRequirements di Refer ensi API Amazon EC2.

```
{
 "SpotOptions": {
  "AllocationStrategy": "price-capacity-optimized"
 },
 "LaunchTemplateConfigs": [{
  "LaunchTemplateSpecification": {
   "LaunchTemplateName": "my-launch-template",
   "Version": "1"
  },
  "Overrides": [{
   "InstanceRequirements": {
    "VCpuCount": {
     "Min": 2
    },
    "MemoryMiB": {
     "Min": 4
    }
   }
  }]
 }],
 "TargetCapacitySpecification": {
  "TotalTargetCapacity": 20,
  "DefaultTargetCapacityType": "spot"
 },
 "Type": "instant"
}
```

# Contoh 12: Meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas

Contoh berikut mengonfigurasi Armada `instant` EC2 untuk meluncurkan instans di beberapa jenis Reservasi Kapasitas — Reservasi Kapasitas, Blok On-Demand Kapasitas untuk ML, dan Reservasi Kapasitas yang dapat diinterupsi — menggunakan Grup Sumber Daya Reservasi Kapasitas tunggal. Anda menentukan preferensi ini dalam `ReservedCapacityOptions` struktur, dan mengatur `DefaultTargetCapacityType` ke`reserved-capacity`.

Dalam contoh berikut:

- `ReservationTypes`— Daftar tipe Reservasi Kapasitas yang diurutkan untuk ditargetkan. EC2 Fleet bekerja melalui tipe dalam urutan ini:`on-demand-capacity-reservation`, lalu`capacity-block`, lalu`interruptible-capacity-reservation`.
- `AllocationStrategy`— Setel `prioritized` untuk memprioritaskan jenis instans dalam setiap jenis Reservasi Kapasitas sesuai dengan urutan penggantian tipe instans Anda.
- `ReservedCapacityFallbackOptions`— `MarketTypes` disetel ke`["on-demand"]`, sehingga armada meluncurkan On-Demand Instans untuk kapasitas target apa pun yang tidak dapat dipenuhi dari kapasitas cadangan. Jika Anda menghilangkan ini, armada tidak akan kembali ke jenis pasar lainnya setelah jenis Reservasi Kapasitas yang ditentukan habis.

Template peluncuran menargetkan Grup Sumber Daya Reservasi Kapasitas. Saat Anda menggunakan`ReservedCapacityOptions`, Anda tidak perlu menyetel template peluncuran — Armada EC2 menyetel ini secara otomatis untuk setiap jenis Reservasi Kapasitas. `MarketType`

```
{
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-cr-resource-group-template",
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "p5.48xlarge",
                    "AvailabilityZone": "us-east-1a",
                    "Priority": 1
                },
```

```
                {
                    "InstanceType": "p4d.48xlarge",
                    "AvailabilityZone": "us-east-1b",
                    "Priority": 2
                }
            ]
        }
    ],
    "TargetCapacitySpecification": {
        "TotalTargetCapacity": 100,
        "DefaultTargetCapacityType": "reserved-capacity"
    },
    "ReservedCapacityOptions": {
        "ReservationTypes": ["on-demand-capacity-reservation", "capacity-block",
 "interruptible-capacity-reservation"],
        "ReservedCapacityFallbackOptions": {
            "MarketTypes": ["on-demand"]
        },
        "AllocationStrategy": "prioritized"
    },
    "Type": "instant"
}
```

Untuk tutorial langkah demi langkah, lihatTutorial: Mengkonfigurasi Armada EC2 Anda untuk meluncurkan instans ke beberapa jenis Reservasi Kapasitas menggunakan Grup Sumber Daya Reservasi Kapasitas.

## Contoh konfigurasi CLI Spot Fleet

Anda dapat menentukan konfigurasi Spot Fleet dalam file JSON, dan kemudian mereferensikan file tersebut menggunakan AWS CLI perintah request-spot-fleet untuk membuat armada Anda, sebagai berikut:

```
aws ec2 request-spot-fleet --spot-fleet-request-config file://file_name.json
```

Contoh berikut menggambarkan konfigurasi peluncuran untuk berbagai kasus penggunaan Spot Fleet. Untuk informasi selengkapnya tentang parameter konfigurasi, lihat request- spot-fleet. Untuk informasi selengkapnya tentang membuat Spot Fleet, lihatMembuat Armada Spot.

> **Note**
>
> Untuk Armada Spot, Anda tidak dapat menentukan ID antarmuka jaringan di templat peluncuran atau spesifikasi peluncuran. Pastikan Anda menghilangkan parameter `NetworkInterfaceID` di templat peluncuran atau spesifikasi peluncuran.

Contoh


- Contoh 1: Luncurkan Instans Spot menggunakan Zona Ketersediaan dengan harga terendah di Wilayah
- Contoh 2: Luncurkan Instans Spot menggunakan Availability Zone dengan harga terendah dari daftar
- Contoh 3: Luncurkan Instans Spot menggunakan jenis instans dengan harga terendah dari daftar
- Contoh 4. Menimpa harga untuk permintaan
- Contoh 5: Meluncurkan Armada Spot menggunakan strategi alokasi yang terdiversifikasi
- Contoh 6: Meluncurkan Armada Spot menggunakan pembobotan instans
- Contoh 7: Luncurkan Armada Spot menggunakan On-Demand kapasitas
- Contoh 8: Mengonfigurasikan Penyeimbangan Ulang Kapasitas untuk meluncurkan Instans Spot pengganti
- Contoh 9: Meluncurkan Instans Spot dalam armada yang dioptimalkan kapasitas
- Contoh 10: Meluncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas
- Contoh 11: Luncurkan Instans Spot di harga CapacityOptimized armada
- Contoh 12: Mengonfigurasi pemilihan tipe instans berbasis atribut


## Contoh 1: Luncurkan Instans Spot menggunakan Zona Ketersediaan dengan harga terendah di Wilayah

Contoh berikut menentukan spesifikasi peluncuran tunggal tanpa Zona Ketersediaan atau subnet. Armada Spot meluncurkan instans di Zona Ketersediaan dengan harga terendah yang memiliki subnet default. Harga yang Anda bayar tidak melebihi On-Demand harga.

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
```

```
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "m3.medium",
          "IamInstanceProfile": {
              "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
          }
      }
  ]
}
```

## Contoh 2: Luncurkan Instans Spot menggunakan Availability Zone dengan harga terendah dari daftar

Contoh berikut menentukan dua spesifikasi peluncuran dengan Zona Ketersediaan atau subnet yang berbeda, tetapi dengan tipe instans dan AMI yang sama.

Zona Ketersediaan

Armada Spot meluncurkan instans di subnet default Zona Ketersediaan dengan harga terendah yang Anda tentukan.

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "m3.medium",
          "Placement": {
```

```
            "AvailabilityZone": "us-west-2a, us-west-2b"
        },
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
        }
    }
  ]
}
```

Subnet

Anda dapat menentukan subnet default atau subnet non-default, dan subnet non-default dapat berasal dari VPC default atau VPC non-default. Layanan Spot meluncurkan instans di subnet mana pun yang berada di Zona Ketersediaan dengan harga terendah.

Anda tidak dapat menentukan subnet yang berbeda dari Zona Ketersediaan yang sama dalam permintaan Armada Spot.

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
    {
        "ImageId": "ami-1a2b3c4d",
        "KeyName": "my-key-pair",
        "SecurityGroups": [
            {
                "GroupId": "sg-1a2b3c4d"
            }
        ],
        "InstanceType": "m3.medium",
        "SubnetId": "subnet-a61dafcf, subnet-65ea5f08",
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
        }
    }
  ]
}
```

Jika instans diluncurkan dalam VPC default, instans akan menerima alamat IPv4 publik secara default. Jika instans diluncurkan dalam VPC non-default, instans tidak menerima alamat IPv4 publik secara default. Gunakan antarmuka jaringan dalam spesifikasi peluncuran untuk menetapkan alamat

IPv4 publik ke instans yang diluncurkan di VPC non-default. Saat Anda menentukan antarmuka jaringan, Anda harus menyertakan ID subnet dan ID grup keamanan menggunakan antarmuka jaringan.

```
  ...
      {
          "ImageId": "ami-1a2b3c4d",
          "KeyName": "my-key-pair",
          "InstanceType": "m3.medium",
          "NetworkInterfaces": [
              {
                  "DeviceIndex": 0,
                  "SubnetId": "subnet-1a2b3c4d",
                  "Groups": [ "sg-1a2b3c4d" ],
                  "AssociatePublicIpAddress": true
              }
          ],
          "IamInstanceProfile": {
              "Arn": "arn:aws:iam::123456789012:instance-profile/my-iam-role"
          }
      }
  ...
```

## Contoh 3: Luncurkan Instans Spot menggunakan jenis instans dengan harga terendah dari daftar

Contoh berikut menentukan dua konfigurasi peluncuran dengan tipe instans yang berbeda, tetapi dengan AMI dan Zona Ketersediaan atau subnet yang sama. Armada Spot meluncurkan instans menggunakan tipe instans yang ditentukan dengan harga terendah.

Zona Ketersediaan

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
```

```
            }
        ],
        "InstanceType": "c5.4xlarge",
        "Placement": {
          "AvailabilityZone": "us-west-2b"
        }
    },
    {
        "ImageId": "ami-1a2b3c4d",
        "SecurityGroups": [
            {
                "GroupId": "sg-1a2b3c4d"
            }
        ],
        "InstanceType": "r3.8xlarge",
        "Placement": {
            "AvailabilityZone": "us-west-2b"
        }
    }
  ]
}
```

### Subnet

```
{
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
              }
          ],
          "InstanceType": "c5.4xlarge",
          "SubnetId": "subnet-1a2b3c4d"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "SecurityGroups": [
              {
                  "GroupId": "sg-1a2b3c4d"
```

```
                }
            ],
            "InstanceType": "r3.8xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        }
    ]
}
```

## Contoh 4. Menimpa harga untuk permintaan

Kami menyarankan Anda menggunakan harga maksimum default, yaitu On-Demand harga. Jika Anda memilih, Anda dapat menentukan harga maksimum untuk permintaan armada dan harga maksimum untuk spesifikasi peluncuran individu.

Contoh berikut menentukan harga maksimum untuk permintaan armada dan harga maksimum untuk dua dari tiga spesifikasi peluncuran. Harga maksimum permintaan armada digunakan untuk spesifikasi peluncuran apa pun yang tidak menentukan harga maksimum. Armada Spot meluncurkan instans menggunakan tipe instans dengan harga terendah.

Zona Ketersediaan

```
{
  "SpotPrice": "1.00",
  "TargetCapacity": 30,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "SpotPrice": "0.10"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.4xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "SpotPrice": "0.20"
      },
```

```
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.8xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          }
      }
    ]
}
```

Subnet

```
{
  "SpotPrice": "1.00",
  "TargetCapacity": 30,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.2xlarge",
          "SubnetId": "subnet-1a2b3c4d",
          "SpotPrice": "0.10"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.4xlarge",
          "SubnetId": "subnet-1a2b3c4d",
          "SpotPrice": "0.20"
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.8xlarge",
          "SubnetId": "subnet-1a2b3c4d"
      }
  ]
}
```

## Contoh 5: Meluncurkan Armada Spot menggunakan strategi alokasi yang terdiversifikasi

Contoh berikut menggunakan strategi alokasi `diversified`. Spesifikasi peluncuran memiliki tipe instans yang berbeda, tetapi AMI dan Zona Ketersediaan atau subnet yang sama. Armada Spot

mendistribusikan 30 instans di tiga spesifikasi peluncuran, sehingga terdapat 10 instans untuk setiap tipe. Untuk informasi selengkapnya, lihat Gunakan strategi alokasi untuk menentukan bagaimana EC2 Fleet atau Spot Fleet memenuhi Spot dan kapasitas On-Demand.

Zona Ketersediaan

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 30,
  "AllocationStrategy": "diversified",
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c4.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          }
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "m3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          }
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "r3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          }
      }
  ]
}
```

Subnet

```
{
    "SpotPrice": "0.70",
    "TargetCapacity": 30,
    "AllocationStrategy": "diversified",
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
```

```
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c4.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "m3.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        }
    ]
}
```

Praktik terbaik untuk meningkatkan peluang permintaan spot dapat dipenuhi oleh kapasitas EC2 jika terjadi pemadaman di salah satu Zona Ketersediaan adalah dengan melakukan diversifikasi di seluruh zona. Untuk skenario ini, sertakan setiap Zona Ketersediaan yang tersedia untuk Anda dalam spesifikasi peluncuran. Selain itu, alih-alih menggunakan subnet yang sama setiap kalinya, gunakan tiga subnet unik (masing-masing memetakan ke zona yang berbeda).

Zona Ketersediaan

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 30,
  "AllocationStrategy": "diversified",
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c4.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2a"
          }
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "m3.2xlarge",
```

```
            "Placement": {
                "AvailabilityZone": "us-west-2b"
            }
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "Placement": {
                "AvailabilityZone": "us-west-2c"
            }
        }
    ]
}
```

### Subnet

```
{
    "SpotPrice": "0.70",
    "TargetCapacity": 30,
    "AllocationStrategy": "diversified",
    "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
    "LaunchSpecifications": [
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "c4.2xlarge",
            "SubnetId": "subnet-1a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "m3.2xlarge",
            "SubnetId": "subnet-2a2b3c4d"
        },
        {
            "ImageId": "ami-1a2b3c4d",
            "InstanceType": "r3.2xlarge",
            "SubnetId": "subnet-3a2b3c4d"
        }
    ]
}
```

# Contoh 6: Meluncurkan Armada Spot menggunakan pembobotan instans

Contoh berikut menggunakan pembobotan instans, yang berarti harga adalah per unit jam, bukan per jam instans. Setiap konfigurasi peluncuran mencantumkan tipe instans yang berbeda dan bobot yang berbeda. Armada Spot memilih tipe instans dengan harga terendah per unit jam. Armada Spot menghitung jumlah Instans Spot yang akan diluncurkan dengan membagi kapasitas target dengan bobot instans. Jika hasilnya bukan bilangan bulat, Armada Spot akan membulatkannya ke bilangan bulat berikutnya, sehingga ukuran armada Anda tidak berada di bawah kapasitas targetnya.

Jika permintaan `r3.2xlarge` berhasil, Spot akan menyediakan 4 instans ini. Bagilah 20 dengan 6 untuk total 3,33 instans, lalu bulatkan menjadi 4 instans.

Jika permintaan `c3.xlarge` berhasil, Spot akan menyediakan 7 instans ini. Bagilah 20 dengan 3 untuk total 6,66 instans, lalu bulatkan menjadi 7 instans.

Untuk informasi selengkapnya, lihat Gunakan pembobotan instans untuk mengelola biaya dan kinerja Armada EC2 atau Spot Fleet.

Zona Ketersediaan

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "r3.2xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "WeightedCapacity": 6
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.xlarge",
          "Placement": {
              "AvailabilityZone": "us-west-2b"
          },
          "WeightedCapacity": 3
      }
    ]
```

```
}
```

Subnet

```
{
  "SpotPrice": "0.70",
  "TargetCapacity": 20,
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "LaunchSpecifications": [
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "r3.2xlarge",
          "SubnetId": "subnet-1a2b3c4d",
          "WeightedCapacity": 6
      },
      {
          "ImageId": "ami-1a2b3c4d",
          "InstanceType": "c3.xlarge",
          "SubnetId": "subnet-1a2b3c4d",
          "WeightedCapacity": 3
      }
  ]
}
```

## Contoh 7: Luncurkan Armada Spot menggunakan On-Demand kapasitas

Untuk memastikan bahwa Anda selalu memiliki kapasitas instans, Anda dapat menyertakan permintaan On-Demand kapasitas dalam permintaan Spot Fleet Anda. Jika ada kapasitas, On-Demand permintaan selalu terpenuhi. Keseimbangan kapasitas target akan terpenuhi sebagai Spot jika terdapat kapasitas dan ketersediaan.

Contoh berikut menentukan kapasitas target yang diinginkan sebagai 10, di mana 5 harus On-Demand kapasitas. Kapasitas spot tidak ditentukan; itu tersirat dalam keseimbangan kapasitas target dikurangi On-Demand kapasitas. Amazon EC2 meluncurkan 5 unit kapasitas as On-Demand, dan 5 unit kapasitas (10-5=5) sebagai Spot jika ada kapasitas dan ketersediaan Amazon EC2 yang tersedia.

```
{
  "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-role",
  "AllocationStrategy": "lowestPrice",
  "TargetCapacity": 10,
```

```
  "SpotPrice": null,
  "ValidFrom": "2018-04-04T15:58:13Z",
  "ValidUntil": "2019-04-04T15:58:13Z",
  "TerminateInstancesWithExpiration": true,
  "LaunchSpecifications": [],
  "Type": "maintain",
  "OnDemandTargetCapacity": 5,
  "LaunchTemplateConfigs": [
    {
      "LaunchTemplateSpecification": {
        "LaunchTemplateId": "lt-0dbb04d4a6cca5ad1",
        "Version": "2"
      },
      "Overrides": [
        {
          "InstanceType": "t2.medium",
          "WeightedCapacity": 1,
          "SubnetId": "subnet-d0dc51fb"
        }
      ]
    }
  ]
}
```

## Contoh 8: Mengonfigurasikan Penyeimbangan Ulang Kapasitas untuk meluncurkan Instans Spot pengganti

Contoh berikut mengonfigurasi Armada Spot untuk meluncurkan Instans Spot pengganti saat Amazon EC2 memancarkan rekomendasi penyeimbangan ulang untuk Instans Spot di armada. Untuk mengonfigurasi penggantian otomatis Instans Spot, untuk `ReplacementStrategy`, tentukan `launch-before-terminate`. Untuk mengonfigurasi waktu tunda dari peluncuran Instans Spot pengganti baru ke penghapusan otomatis Instans Spot lama, untuk `termination-delay`, tentukan nilai dalam hitungan detik. Untuk informasi selengkapnya, lihat Opsi konfigurasi.

> **Note**
>
> Sebaiknya gunakan `launch-before-terminate` hanya jika Anda dapat memprediksi lamanya prosedur pematian instans Anda akan selesai. Hal ini memastikan bahwa instans lama diakhiri hanya setelah prosedur pematian selesai. Anda dikenai biaya untuk semua instans saat semuanya berjalan.

Efektivitas strategi Penyeimbangan Ulang Kapasitas bergantung pada jumlah kolam kapasitas Spot yang ditentukan dalam permintaan Armada Spot. Sebaiknya konfigurasikan armada dengan set tipe instans dan Zona Ketersediaan yang beragam, dan untuk `AllocationStrategy`, tentukan `capacityOptimized`. Untuk informasi selengkapnya tentang hal-hal yang harus Anda pertimbangkan saat mengonfigurasi Armada Spot untuk Penyeimbangan Ulang Kapasitas, lihat Gunakan Penyeimbangan Kembali Kapasitas di Armada EC2 dan Spot Fleet untuk menggantikan Instans Spot yang berisiko.

```
{
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "capacityOptimized",
        "IamFleetRole": "arn:aws:iam::123456789012:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateName": "LaunchTemplate",
                    "Version": "1"
                },
                 "Overrides": [
                       {
                           "InstanceType": "c3.large",
                           "WeightedCapacity": 1,
                           "Placement": {
                               "AvailabilityZone": "us-east-1a"
                           }
                       },
                       {
                           "InstanceType": "c4.large",
                           "WeightedCapacity": 1,
                           "Placement": {
                               "AvailabilityZone": "us-east-1a"
                           }
                       },
                       {
                           "InstanceType": "c5.large",
                           "WeightedCapacity": 1,
                           "Placement": {
                               "AvailabilityZone": "us-east-1a"
                           }
                       }
                ]
```

```
          }
    ],
        "TargetCapacity": 5,
        "SpotMaintenanceStrategies": {
            "CapacityRebalance": {
                "ReplacementStrategy": "launch-before-terminate",
                "TerminationDelay": "720"
            }
        }
    }
}
```

## Contoh 9: Meluncurkan Instans Spot dalam armada yang dioptimalkan kapasitas

Contoh berikut menunjukkan cara mengonfigurasi Armada Spot dengan strategi alokasi Spot yang mengoptimalkan kapasitas. Untuk mengoptimalkan kapasitas, Anda harus mengatur `AllocationStrategy` ke `capacityOptimized`.

Pada contoh berikut ini, tiga spesifikasi peluncuran menentukan tiga kolam kapasitas Spot. Kapasitas target adalah 50 Instans Spot. Armada Spot berupaya meluncurkan 50 Instans Spot ke kolam kapasitas Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan.

```
{
    "TargetCapacity": "50",
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "capacityOptimized",
    },
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "r4.2xlarge",
                    "AvailabilityZone": "us-west-2a"
                },
                {
                    "InstanceType": "m4.2xlarge",
                    "AvailabilityZone": "us-west-2b"
```

```
                },
                {
                    "InstanceType": "c5.2xlarge",
                    "AvailabilityZone": "us-west-2b"
                }
            ]
        }
    ]
}
```

## Contoh 10: Meluncurkan Instans Spot dalam armada yang dioptimalkan kapasitas dengan prioritas

Contoh berikut menunjukkan cara mengonfigurasi Armada Spot dengan strategi alokasi Spot yang mengoptimalkan kapasitas sambil menggunakan prioritas dengan upaya terbaik.

Jika menggunakan strategi alokasi `capacityOptimizedPrioritized`, Anda dapat menggunakan parameter `Priority` untuk menentukan prioritas kolam kapasitas Spot, yaitu makin rendah angkanya, makin tinggi prioritasnya. Anda juga dapat mengatur prioritas yang sama untuk beberapa kolam kapasitas Spot jika Anda menginginkannya setara. Jika Anda tidak menetapkan prioritas, kolam akan dianggap yang terakhir dalam hal prioritas.

Untuk memprioritaskan kolam kapasitas Spot, Anda harus mengatur `AllocationStrategy` ke `capacityOptimizedPrioritized`. Armada Spot akan mengoptimalkan kapasitas terlebih dahulu, tetapi akan mempertimbangkan prioritas dengan upaya terbaik (misalnya, jika mempertimbangkan prioritas tidak akan secara signifikan memengaruhi kemampuan Armada Spot untuk menyediakan kapasitas optimal). Ini adalah pilihan opsi yang bagus untuk beban kerja di mana kemungkinan gangguan harus diminimalkan dan preferensi untuk tipe instans tertentu menjadi penting.

Pada contoh berikut ini, tiga spesifikasi peluncuran menentukan tiga kolam kapasitas Spot. Setiap kolam diprioritaskan, yaitu makin rendah jumlahnya, makin tinggi prioritasnya. Kapasitas target adalah 50 Instans Spot. Armada Spot berupaya meluncurkan 50 Instans Spot ke dalam kolam kapasitas Spot dengan prioritas tertinggi menggunakan upaya terbaik, tetapi mengoptimalkan kapasitas terlebih dahulu.

```
{
    "TargetCapacity": "50",
    "SpotFleetRequestConfig": {
        "AllocationStrategy": "capacityOptimizedPrioritized"
```

```
    },
    "LaunchTemplateConfigs": [
        {
            "LaunchTemplateSpecification": {
                "LaunchTemplateName": "my-launch-template",
                "Version": "1"
            },
            "Overrides": [
                {
                    "InstanceType": "r4.2xlarge",
                   "Priority": 1,
                    "AvailabilityZone": "us-west-2a"
                },
                {
                          "InstanceType": "m4.2xlarge",
                          "Priority": 2,
                          "AvailabilityZone": "us-west-2b"
                },
                {
                          "InstanceType": "c5.2xlarge",
                          "Priority": 3,
                          "AvailabilityZone": "us-west-2b"
                }
            ]
        }
    ]
}
```

## Contoh 11: Luncurkan Instans Spot di harga CapacityOptimized armada

Contoh berikut menunjukkan cara mengonfigurasi Armada Spot dengan strategi alokasi Spot yang mengoptimalkan kapasitas dan harga terendah. Untuk mengoptimalkan kapasitas sambil mempertimbangkan harga, Anda harus mengatur Spot `AllocationStrategy` ke `priceCapacityOptimized`.

Pada contoh berikut ini, tiga spesifikasi peluncuran menentukan tiga kolam kapasitas Spot. Kapasitas target adalah 50 Instans Spot. Armada Spot berupaya meluncurkan 50 Instans Spot ke kolam kapasitas Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan sekaligus memilih kolam yang memiliki harga terendah.

```
{
    "SpotFleetRequestConfig": {
```

```
        "AllocationStrategy": "priceCapacityOptimized",
        "OnDemandAllocationStrategy": "lowestPrice",
        "ExcessCapacityTerminationPolicy": "default",
        "IamFleetRole": "arn:aws:iam::111111111111:role/aws-ec2-spot-fleet-tagging-
role",
        "LaunchTemplateConfigs": [
            {
                "LaunchTemplateSpecification": {
                    "LaunchTemplateId": "lt-0123456789example",
                    "Version": "1"
                },
                "Overrides": [
                     {
                           "InstanceType": "r4.2xlarge",
                           "AvailabilityZone": "us-west-2a"
                      },
                      {
                           "InstanceType": "m4.2xlarge",
                           "AvailabilityZone": "us-west-2b"
                      },
                      {
                           "InstanceType": "c5.2xlarge",
                           "AvailabilityZone": "us-west-2b"
                      }
                ]
            }
        ],
        "TargetCapacity": 50,
        "Type": "request"
    }
}
```

## Contoh 12: Mengonfigurasi pemilihan tipe instans berbasis atribut

Contoh berikut menunjukkan cara mengonfigurasi Armada Spot untuk menggunakan pemilihan tipe instans berbasis atribut untuk mengidentifikasi tipe instans. Untuk menentukan atribut instans yang diperlukan, Anda menentukan atribut dalam struktur `InstanceRequirements`.

Pada contoh berikut ini, dua atribut instans ditentukan:

- `VCpuCount` – Minimum 2 vCPU ditentukan. Karena tidak ada jumlah maksimum yang ditentukan, maka tidak ada batas maksimum.

- `MemoryMiB` – Minimum 4 MiB memori ditentukan. Karena tidak ada jumlah maksimum yang ditentukan, maka tidak ada batas maksimum.

Setiap tipe instans yang memiliki 2 atau lebih VCPU dan 4 MiB atau lebih memori akan diidentifikasi. Namun, perlindungan harga dan strategi alokasi mungkin akan mengecualikan beberapa tipe instans jika Armada Spot menyediakan armada.

Untuk daftar dan deskripsi semua atribut yang mungkin dapat Anda tentukan, lihat InstanceRequirements di Refer ensi API Amazon EC2.

```
{
 "AllocationStrategy": "priceCapacityOptimized",
 "TargetCapacity": 20,
 "Type": "request",
 "LaunchTemplateConfigs": [{
  "LaunchTemplateSpecification": {
   "LaunchTemplateName": "my-launch-template",
   "Version": "1"
  },
  "Overrides": [{
   "InstanceRequirements": {
    "VCpuCount": {
     "Min": 2
    },
    "MemoryMiB": {
     "Min": 4
    }
   }
  }]
 }]
}
```

# Kuota untuk Armada EC2 dan Armada Spot

Anda Akun AWS memiliki kuota default, sebelumnya disebut sebagai batas, untuk setiap AWS layanan. Kecuali dinyatakan lain, setiap kuota adalah Region-specific.

Kuota Amazon EC2 yang biasa berlaku untuk instans yang diluncurkan oleh Armada EC2 atau Armada Spot, seperti batas Instans Spot dan batas volume.

Selain itu, Anda Akun AWS memiliki kuota berikut terkait dengan Armada EC2 dan Spot Fleet:

| Deskripsi kuota | Kuota |
| --- | --- |
| Jumlah Armada EC2 dan Armada Spot per Wilayah jenis `maintain` dan `request` diactive,, `deleted_running` dan negara bagian `cancelled_running` | 1.000[1] [2][3] |
| Jumlah Armada EC2 jenis `instant` | Tidak terbatas. |
| Jumlah kumpulan kapasitas Spot (kombinasi unik dari jenis instans dan subnet) untuk Armada EC2 dan Armada Spot jenis dan `maintain` `request` | 300[1] |
| Jumlah kumpulan kapasitas Spot (kombinasi unik dari jenis instans dan subnet) untuk jenis Armada EC2 `instant` | Tidak terbatas. |
| Ukuran data pengguna dalam spesifikasi peluncuran | 16 KB [2] |
| Kapasitas target per Armada EC2 atau Armada Spot | 10,000 |
| Jumlah Reservasi Kapasitas per permintaan Armada EC2 saat menggunakan `ReservedCapacityOptions` | 5.000 m² |
| Kapasitas target di semua Armada EC2 dan Armada Spot di suatu Wilayah | 100.000 [1] |
| Permintaan Armada EC2 atau permintaan Armada Spot tidak dapat menjangkau Wilayah. | |
| Permintaan Armada EC2 atau permintaan Armada Spot tidak dapat menjangkau subnet yang berbeda dari Zona Ketersediaan yang sama. | |

[1] Kuota ini berlaku untuk Armada EC2 dan Armada Spot Anda.

[2] Kuota ini merupakan kuota hard. Anda tidak dapat meminta kenaikan untuk kuota ini.

[3] Setelah Anda menghapus Armada EC2 atau membatalkan permintaan Armada Spot, dan jika Anda menentukan bahwa armada tidak boleh mengakhiri Instans Spot-nya saat Anda menghapus atau membatalkan permintaan, permintaan armada akan memasuki status `deleted_running` (Armada EC2) atau `cancelled_running` (Armada Spot) dan instans terus berjalan hingga terinterupsi atau Anda mengakhirinya secara manual. Jika Anda menghentikan instans, permintaan armada akan memasuki status `deleted_terminating` (Armada EC2) atau `cancelled_terminating` (Armada Spot) dan tidak dihitung dalam kuota ini. Untuk informasi selengkapnya, lihat Menghapus permintaan Armada EC2 dan instans di armada dan Membatalkan (menghapus) permintaan Spot Fleet.

## Meminta peningkatan kuota untuk kapasitas target

Jika Anda membutuhkan lebih dari kuota default untuk kapasitas target, Anda dapat meminta peningkatan kuota.

Untuk meminta peningkatan kuota pada kapasitas target

1. Buka formulir Dukungan Center Create case.
2. Pilih Peningkatan batas layanan.
3. Untuk Tipe batas, pilih Armada EC2.
4. Untuk  Wilayah, pilih AWS Wilayah tempat meminta kenaikan kuota.
5. Untuk Batas, pilih Kapasitas Armada Target per Armada (dalam unit) atau Kapasitas Armada Target per Wilayah (dalam unit), bergantung pada kuota yang ingin Anda tingkatkan.
6. Untuk Nilai batas baru, masukkan nilai kuota baru.
7. Untuk meminta peningkatan kuota lain, pilih Tambahkan permintaan lain, dan ulangi Langkah 4–6.
8. Untuk Deskripsi kasus penggunaan, masukkan alasan Anda meminta peningkatan kuota.
9. Di Opsi kontak, tentukan bahasa kontak dan metode kontak pilihan Anda.
10. Pilih Kirim.

# Jaringan di Amazon EC2

Amazon VPC memungkinkan Anda meluncurkan AWS sumber daya, seperti instans Amazon EC2, ke dalam jaringan virtual yang didedikasikan untuk akun AWS Anda, yang dikenal sebagai virtual private cloud (VPC). Saat Anda meluncurkan sebuah instans, Anda dapat memilih subnet dari VPC. Instans dikonfigurasi dengan antarmuka jaringan primer, yang merupakan kartu jaringan virtual logis. Instance menerima alamat IP pribadi utama dari IPv4 alamat subnet, dan ditugaskan ke antarmuka jaringan utama.

Anda dapat mengendalikan apakah instans menerima alamat IP publik dari kumpulan alamat IP publik Amazon. Alamat IP publik dari sebuah instans dikaitkan dengan instans Anda hanya sampai dihentikan atau diakhiri. Jika Anda memerlukan alamat IP publik persisten, Anda dapat mengalokasikan alamat IP Elastis untuk AWS akun Anda dan mengaitkannya dengan instance atau antarmuka jaringan. Alamat IP Elastis tetap terkait dengan AWS akun Anda sampai Anda melepaskannya, dan Anda dapat memindahkannya dari satu instance ke yang lain sesuai kebutuhan. Anda dapat membawa rentang alamat IP Anda sendiri ke akun AWS Anda, di mana muncul sebagai kumpulan alamat, dan kemudian mengalokasikan alamat IP elastis dari kumpulan alamat Anda.

Untuk meningkatkan performa jaringan dan mengurangi latensi, Anda dapat meluncurkan instans dalam grup penempatan. Anda bisa mendapatkan performa paket per detik (PPS) yang jauh lebih tinggi menggunakan jaringan yang ditingkatkan. Anda dapat mempercepat komputasi performa tinggi dan aplikasi machine learning menggunakan Elastic Fabric Adapter (EFA), yang merupakan perangkat jaringan yang dapat Anda pasang ke tipe instans yang didukung.

Fitur

- Wilayah dan Zona
- Pengalamatan IP instans Amazon EC2
- Nama host dan domain instans EC2
- Bawa alamat IP Anda sendiri (BYOIP) ke Amazon EC2
- Alamat Elastic IP
- Antarmuka jaringan elastis
- Bandwidth jaringan instans Amazon EC2
- Jaringan yang disempurnakan di instans Amazon EC2
- Adaptor Kain Elastis untuk AI/ML dan beban kerja HPC di Amazon EC2
- Topologi Amazon EC2

- Grup penempatan untuk instans Amazon EC2 Anda
- Maximum transmission unit (MTU) jaringan untuk instans EC2 Anda
- Virtual private cloud untuk instans EC2
- Jaringan Sekunder

# Wilayah dan Zona

Amazon EC2 dihosting di beberapa lokasi di seluruh dunia. Lokasi ini terdiri dari Wilayah AWS, Zona Ketersediaan, Zona Lokal AWS Outposts, dan Zona Panjang Gelombang.

- Wilayah  adalah wilayah geografis yang terpisah.
- Zona Ketersediaan  adalah beberapa lokasi terisolasi di setiap Wilayah.
- Zona Lokal  memberi Anda kemampuan untuk menempatkan sumber daya, seperti komputasi dan penyimpanan, di beberapa lokasi yang lebih dekat dengan pengguna akhir Anda.
- Zona Panjang  Gelombang memberi Anda kemampuan untuk membangun aplikasi yang memberikan latensi sangat rendah ke perangkat 5G dan pengguna akhir. Wavelength menyebarkan layanan AWS komputasi dan penyimpanan standar ke tepi jaringan 5G operator telekomunikasi.
- AWS Outpostsmenghadirkan AWS layanan asli, infrastruktur, dan model operasi ke hampir semua pusat data, ruang colocation, atau fasilitas lokal.

AWS mengoperasikan pusat data yang canggih dan sangat tersedia. Meskipun jarang terjadi, kegagalan dapat terjadi yang memengaruhi ketersediaan instans yang berada di lokasi yang sama. Jika Anda meng-host semua instans Anda di satu lokasi yang dipengaruhi oleh kegagalan, tidak ada instans Anda yang akan tersedia.

Untuk informasi lebih lanjut, lihat AWS Infrastruktur Global.

Daftar Isi


- Wilayah
- Zona Ketersediaan
- Zona Lokal
- Wavelength Zones
- AWS Outposts

# Wilayah

Setiap Wilayah dirancang untuk diisolasi dari Wilayah lainnya. Ini mencapai toleransi kesalahan dan stabilitas sebesar mungkin.

Saat meluncurkan instance, pilih Wilayah yang menempatkan instance Anda dekat dengan pelanggan tertentu, atau yang memenuhi persyaratan hukum atau persyaratan lain yang Anda miliki. Anda dapat meluncurkan instans di beberapa Wilayah.

Saat melihat sumber daya, Anda hanya melihat sumber daya yang terkait dengan Wilayah yang Anda tentukan. Hal ini karena Wilayah terisolasi dari satu sama lain dan kami tidak mereplikasi sumber daya di seluruh Wilayah secara otomatis.

## Wilayah yang Tersedia

Untuk daftar Wilayah yang tersedia, lihat Wil AWS ayah.

## Titik akhir regional untuk Amazon EC2

Saat Anda bekerja dengan sebuah instans menggunakan antarmuka baris perintah atau tindakan API, Anda harus menentukan titik akhir Wilayah. Untuk informasi selengkapnya tentang Wilayah dan titik akhir untuk Amazon EC2, lihat titik akhir layanan Amazon EC2 di Panduan Pengembang Amazon EC2.

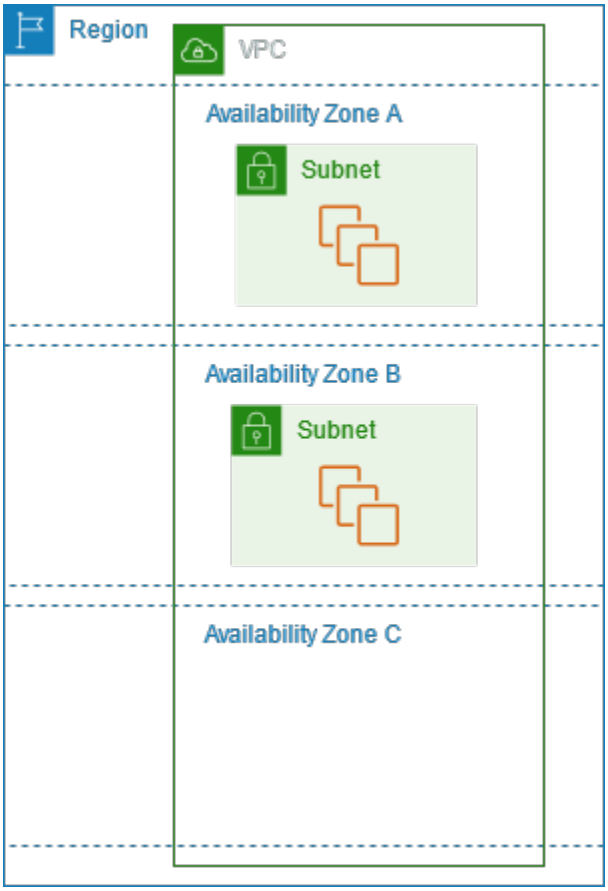
Untuk informasi selengkapnya, lihat Wilayah AWS di Wilayah AWS dan Panduan Pengguna Zona Ketersediaan.

# Zona Ketersediaan

Setiap Wilayah memiliki beberapa lokasi terisolasi yang dikenal sebagai Zona Ketersediaan. Kode untuk Zona Ketersediaan adalah kode Wilayahnya diikuti dengan pengidentifikasi huruf. Misalnya, `us-east-1a`.

Dengan meluncurkan instans EC2 di beberapa Zona Ketersediaan, Anda dapat melindungi aplikasi Anda dari kegagalan satu lokasi di Wilayah.

Diagram berikut menggambarkan beberapa Zona Ketersediaan di suatu Wil AWS ayah. Zona Ketersediaan A dan Zona Ketersediaan B masing-masing memiliki satu subnet, dan setiap subnet memiliki instans EC2. Zona Ketersediaan C tidak memiliki subnet, oleh karena itu Anda tidak dapat meluncurkan instans ke Zona Ketersediaan ini.




Untuk informasi selengkapnya, lihat Virtual private cloud untuk instans EC2.

## Zona Ketersediaan menurut Wilayah

Untuk daftar Zona Ketersediaan menurut Wilayah, lihat Zona AWS Ketersediaan.

## Instans di Zona Ketersediaan

Saat meluncurkan instance, Anda memilih Wilayah dan cloud pribadi virtual (VPC). Kemudian, Anda dapat memilih subnet dari salah satu Zona Ketersediaan atau membiarkan kami memilih subnet untuk Anda. Saat meluncurkan instans awal, sebaiknya Anda mengizinkan kami memilih Zona Ketersediaan untuk Anda berdasarkan kesehatan sistem dan kapasitas yang tersedia. Jika Anda meluncurkan instans tambahan, tentukan Zona Ketersediaan hanya jika instans baru Anda harus dekat atau terpisah dari instans yang sudah ada.

Jika Anda mendistribusikan instans di beberapa Zona Ketersediaan dan instans gagal, Anda dapat merancang aplikasi Anda sehingga instance di Zona Ketersediaan lain menangani permintaan sebagai gantinya.

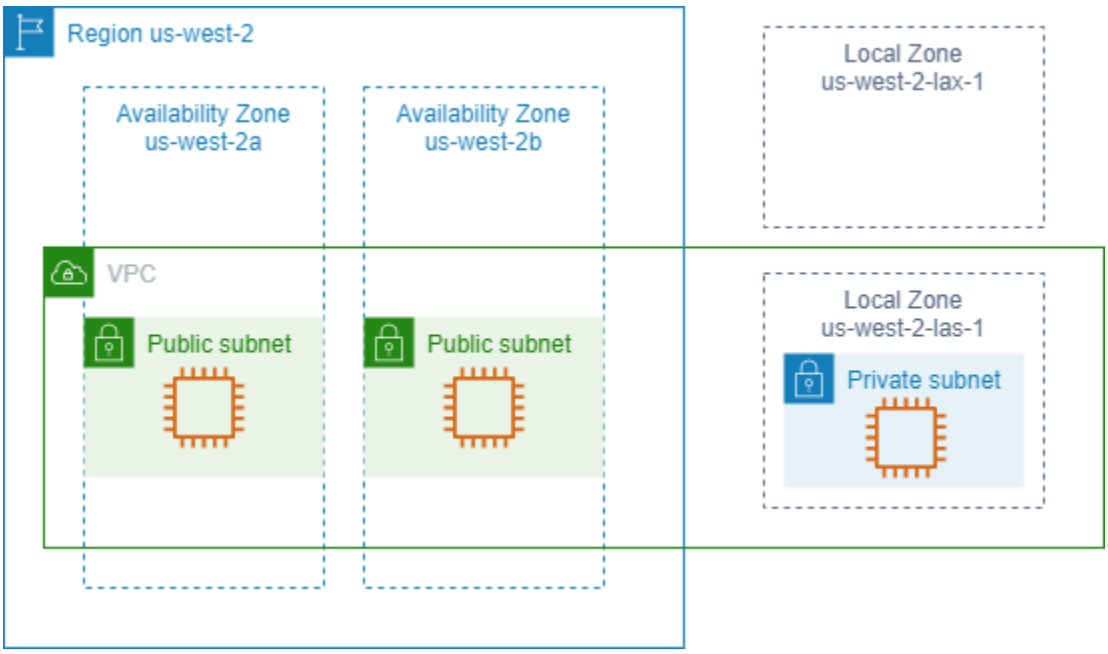
Untuk informasi selengkapnya, lihat Zona Keter AWS sediaan di Wilayah AWS dan Panduan Pengguna Zona Ketersediaan.

## Zona Lokal

Zona Lokal adalah perpanjangan dari Wil AWS ayah dalam kedekatan geografis dengan pengguna Anda. Zona Lokal memiliki koneksi sendiri ke internet dan dukungan Direct Connect, sehingga sumber daya yang dibuat di Zona Lokal dapat melayani pengguna lokal dengan komunikasi latensi rendah. Untuk informasi selengkapnya, lihat Apa itu AWS Zona Lokal? dalam Panduan AWS Pengguna Zona Lokal.

Kode untuk Local Zones adalah kode Wilayah diikuti oleh pengidentifikasi yang menunjukkan lokasi fisiknya. Misalnya, `us-west-2-lax-1` di Los Angeles.

Diagram berikut menggambarkan AWS Wilayah`us-west-2`, dua Zona Ketersediaan, dan dua Zona Lokalnya. VPC mencakup Zona Ketersediaan dan salah satu Local Zones. Setiap zona di VPC memiliki satu subnet, dan setiap subnet memiliki instans.




### Local Zones yang Tersedia

Untuk daftar Zona Lokal yang tersedia, lihat Zona Lokal Ter sedia di Panduan AWS Pengguna Zona Lokal. Untuk daftar Zona Lokal yang diumumkan, lihat Lokasi Zona AWS Lokal.

### Contoh di Zona Lokal

Untuk menggunakan Local Zones, Anda harus mengaktifkannya terlebih dahulu. Kemudian, buat subnet di Zona Lokal. Anda dapat menentukan subnet Zona Lokal saat meluncurkan instance, yang menempatkannya di subnet Zona Lokal di Zona Lokal.

Saat Anda meluncurkan instance di Zona Lokal, Anda juga mengalokasikan alamat IP dari grup perbatasan jaringan. Grup perbatasan jaringan adalah kumpulan unik Zona Ketersediaan, Zona Lokal, atau Zona Panjang Gelombang yang meng AWS iklankan alamat IP, misalnya,`us-west-2-lax-1a`. Anda dapat mengalokasikan alamat IP berikut dari grup batas jaringan:

- Amazon-provided Alamat IPv4 elastis
- Amazon-provided Alamat IPv6 VPC (hanya tersedia di zona Los Angeles)

Untuk informasi selengkapnya tentang cara meluncurkan instance di Zona Lokal, lihat Mem ulai dengan Zona AWS Lokal di Panduan Pengguna Zona AWS Lokal.

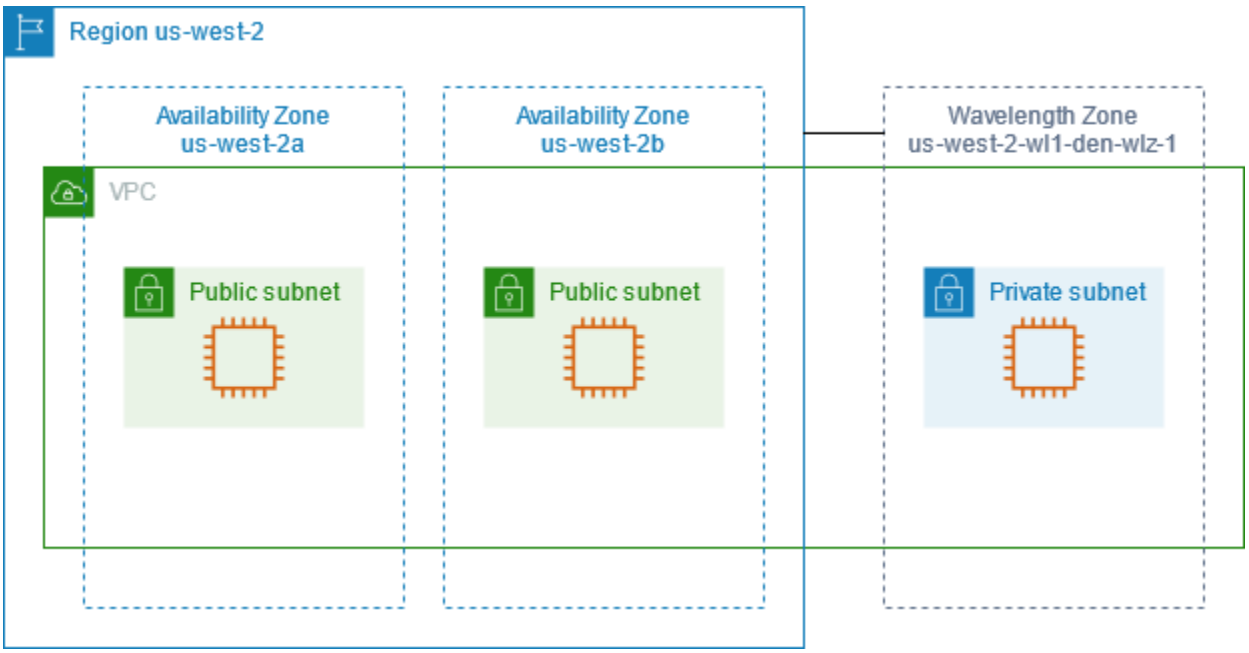
## Wavelength Zones

AWS Wavelength memungkinkan pengembang untuk membangun aplikasi yang memberikan latensi ultra-rendah ke perangkat seluler dan pengguna akhir. Wavelength menyebarkan layanan AWS komputasi dan penyimpanan standar ke tepi jaringan 5G operator telekomunikasi. Pengembang dapat memperluas cloud pribadi virtual (VPC) ke satu atau lebih Zona Panjang Gelombang, dan kemudian menggunakan AWS sumber daya seperti instans Amazon EC2 untuk menjalankan aplikasi yang memerlukan latensi sangat rendah dan koneksi ke AWS layanan di Wilayah.

Zona Panjang Gelombang adalah zona terisolasi di lokasi pembawa tempat infrastruktur panjang gelombang digunakan. Wavelength Zone terikat pada suatu Wilayah. Zona Panjang Gelombang adalah perpanjangan logis dari Wilayah, dan dikelola oleh bidang kontrol di Wilayah.

Kode untuk Wavelength Zone adalah kode Wilayah diikuti oleh pengidentifikasi yang menunjukkan lokasi fisik. Misalnya, `us-east-1-wl1-bos-wlz-1` di Boston.

Diagram berikut menggambarkan Wil AWS ayah`us-west-2`, dua Zona Ketersediaan, dan Zona Panjang Gelombang. VPC mencakup Zona Ketersediaan dan Wavelength Zone. Setiap zona di VPC memiliki satu subnet, dan setiap subnet memiliki instans.




Wavelength Zones tidak tersedia di setiap Wilayah. Untuk informasi tentang Wilayah yang mendukung Wavelength Zones, lihat Wavelength Zones yang Tersedia di Panduan Developer AWS Wavelength .

### Zona Panjang Gelombang yang Tersedia

Untuk daftar Zona Panjang Gelombang yang tersedia, lihat Zona Panjang Gelombang yang Tersedia dalam AWS Wavelength Panduan.

### Contoh di Zona Panjang Gelombang

Untuk menggunakan Wavelength Zone, Anda harus terlebih dahulu memilih Zona. Kemudian, buat subnet di Zona Panjang Gelombang. Anda dapat menentukan subnet Wavelength saat meluncurkan instance. Anda juga mengalokasikan alamat IP operator dari grup batas jaringan, yang merupakan kumpulan unik dari Zona Ketersediaan, Local Zones, atau Wavelength Zone tempat AWS mengiklankan alamat IP, misalnya, `us-east-1-wl1-bos-wlz-1`.

Untuk petunjuk langkah demi langkah untuk meluncurkan instance di Zona Panjang Gelombang, lihat Mem ulai dengan AWS Wavelength di Panduan Peng AWS Wavelength embang.

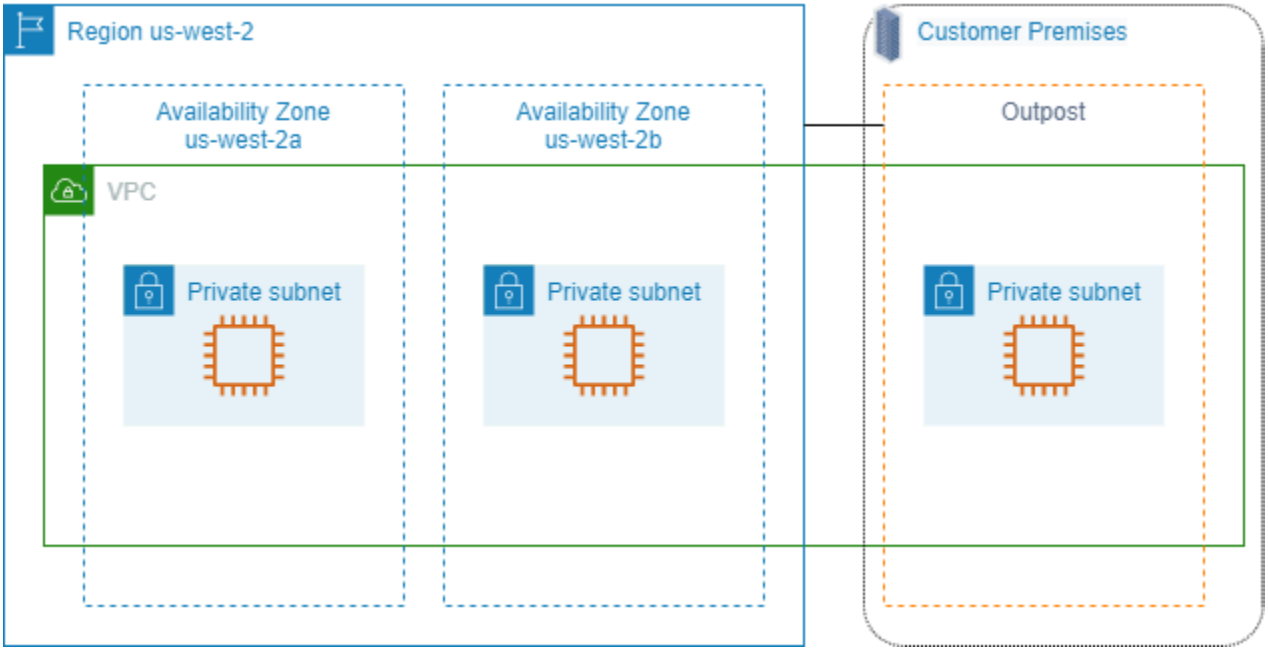
## AWS Outposts

AWS Outposts adalah layanan yang dikelola sepenuhnya yang memperluas AWS infrastruktur, layanan, API, dan alat ke tempat pelanggan. Dengan menyediakan akses lokal ke infrastruktur AWS terkelola, AWS Outposts memungkinkan pelanggan untuk membangun dan menjalankan aplikasi

di tempat menggunakan antarmuka pemrograman yang sama seperti di Wil AWS ayah, sambil menggunakan sumber daya komputasi dan penyimpanan lokal untuk latensi yang lebih rendah dan kebutuhan pemrosesan data lokal.

Outpost adalah kumpulan kapasitas AWS komputasi dan penyimpanan yang digunakan di situs pelanggan. AWS mengoperasikan, memantau, dan mengelola kapasitas ini sebagai bagian dari Wil AWS ayah. Anda dapat membuat subnet di Outpost Anda dan menentukannya saat Anda membuat AWS sumber daya. Instans di subnet Outpost berkomunikasi dengan instans lain di Wil AWS ayah menggunakan alamat IP pribadi, semuanya dalam VPC yang sama.

Diagram berikut menggambarkan Wil AWS ayah`us-west-2`, dua Zona Ketersediaannya, dan Pos Terdepan. VPC mencakup Zona Ketersediaan dan Outpost. Outpost berada di pusat data pelanggan on-premise. Setiap zona di VPC memiliki satu subnet, dan setiap subnet memiliki instans.




## Contoh di Pos Terdepan

Untuk mulai menggunakan AWS Outposts, Anda harus membuat Pos Terdepan dan memesan kapasitas Outpost. AWS Outposts menawarkan dua faktor bentuk, rak pos terdepan dan server pos terdepan. Untuk informasi selengkapnya tentang konfigurasi Outpost, lihat AWS Outposts Keluarga. Setelah peralatan Outpost diinstal, kapasitas komputasi dan penyimpanan tersedia untuk Anda saat meluncurkan instans EC2 di Outpost Anda.

Untuk meluncurkan instans EC2, Anda harus membuat subnet Outpost. Grup keamanan mengontrol lalu lintas masuk dan keluar untuk instans di subnet Outpost, sama seperti yang mereka lakukan untuk instance di subnet Zona Ketersediaan. Untuk menyambung ke instans di subnet Outpost

menggunakan SSH, tentukan pasangan kunci saat Anda meluncurkannya, seperti yang Anda lakukan untuk instance di subnet Availability Zone.

Untuk informasi selengkapnya, lihat Memulai dengan rak Outpost atau Mem ulai dengan server Outpost.

### Volume di rak Pos Terdepan

Jika kapasitas komputasi Outpost Anda berada di rak Outpost, Anda dapat membuat volume EBS di subnet Outpost yang Anda buat. Saat Anda membuat volume, tentukan Amazon Resource Name (ARN) dari Outpost.

Perintah buat volume berikut membuat volume kosong 50 GB di Pos terdepan yang ditentukan.

```
aws ec2 create-volume --availability-zone us-east-2a --outpost-arn arn:aws:outposts:us-east-2:123456789012:outpost/op-03e6fecad652a6138 --size 50
```

Anda dapat secara dinamis mengubah ukuran volume gp2 Amazon EBS Anda tanpa melepaskan mereka. Untuk informasi selengkapnya tentang memodifikasi volume tanpa melepaskannya, lihat Meminta modifikasi pada volume EBS Anda di Panduan Pengguna Amazon EBS.

Sebaiknya batasi volume root untuk instance di rak Outpost hingga 30 GiB atau lebih kecil. Anda dapat menentukan volume data dalam pemetaan perangkat blok dari AMI atau instans untuk menyediakan penyimpanan tambahan. Untuk memangkas blok yang tidak digunakan dari volume boot, lihat Cara Membangun Volume EBS Sparse di Blog Jaringan AWS Mitra.

Kami menyarankan Anda untuk meningkatkan waktu tunggu NVMe untuk volume root. Untuk informasi selengkapnya, lihat batas waktu I/O operasi di Panduan Pengguna Amazon EBS.

### Volume pada server Outposts

Instans pada server Outposts menyediakan volume penyimpanan instans tetapi tidak mendukung volume EBS. Pilih EBS-backed AMI Amazon dengan hanya satu snapshot EBS. Pilih ukuran instance dengan penyimpanan instans yang cukup untuk memenuhi kebutuhan aplikasi Anda. Lihat informasi yang lebih lengkap di Batas volume penyimpanan instans.

## Pengalamatan IP instans Amazon EC2

Amazon EC2 dan Amazon VPC mendukung protokol pengalamatan IPv4 dan IPv6. Secara default, Amazon VPC menggunakan protokol pengalamatan IPv4; Anda tidak dapat menonaktifkan perilaku

ini. Saat Anda membuat VPC, Anda harus menentukan blok CIDR IPv4 (rentang alamat IPv4 privat). Atau, Anda dapat menetapkan blok CIDR IPv6 ke VPC Anda dan menetapkan alamat IPv6 dari blok tersebut ke instans di subnet Anda.

Saat Anda meluncurkan instance EC2, Anda menentukan VPC dan subnet. Instans menerima alamat IPv4 pribadi dari rentang CIDR subnet. Anda dapat secara opsional mengonfigurasi instans Anda dengan alamat IPv4 publik dan alamat IPv6. Jika instance EC2 di VPC yang berbeda berkomunikasi menggunakan alamat IP publik, lalu lintas tetap berada di jaringan global AWS pribadi dan tidak melintasi internet publik.

Daftar Isi


- Alamat IPv4 privat
- Alamat IPv4 publik
- Pengoptimalan alamat IPv4 publik
- Alamat IPv6
- Beberapa alamat IP
- Nama host instans EC2
- Link-local alamat
- Mengelola alamat IPv4 untuk instans EC2 Anda
- Mengelola alamat IPv6 untuk instans EC2 Anda
- Alamat IP sekunder untuk instans EC2
- Konfigurasikan alamat IPv4 pribadi sekunder untuk instans Windows


## Alamat IPv4 privat

Alamat IPv4 privat adalah alamat IP yang tidak dapat dijangkau dengan Internet. Anda dapat menggunakan alamat IPv4 privat untuk komunikasi antara instans dalam VPC yang sama. Untuk informasi selengkapnya tentang standar dan spesifikasi alamat IPv4 privat, lihat RFC 1918. Kami mengalokasikan alamat IPv4 privat ke instans menggunakan DHCP.

> **Note**
>
> Anda dapat membuat VPC dengan blok CIDR yang dapat dirutekan secara publik yang berada di luar rentang alamat IPv4 prvat yang ditentukan dalam RFC 1918. Namun, untuk

keperluan dokumentasi ini, kami merujuk pada alamat IPv4 privat (or 'alamat IP privat') sebagai alamat IP yang berada dalam rentang CIDR IPv4 VPC Anda.

Subnet VPC dapat berupa salah satu tipe dari berikut ini:

- IPv4-only subnet — Anda hanya dapat membuat sumber daya di subnet ini dengan alamat IPv4 yang ditetapkan untuk mereka.
- IPv6-only subnet — Anda hanya dapat membuat sumber daya di subnet ini dengan alamat IPv6 yang ditetapkan untuk mereka.
- Subnet IPv4 dan IPv6 — Anda dapat membuat sumber daya di subnet ini dengan alamat IPv4 atau IPv6 yang ditetapkan untuk mereka.

Saat Anda meluncurkan instance EC2 ke subnet IPv4-only atau dual stack (IPv4 dan IPv6), instans menerima alamat IP pribadi utama dari rentang alamat IPv4 subnet. Untuk informasi selengkapnya, lihat ACL Jaringan di Panduan Pengguna Amazon VPC. Jika Anda tidak menentukan alamat IP privat primer saat Anda meluncurkan instans, kami akan memilih alamat IP yang tersedia di rentang subnet IPv4 untuk Anda. Setiap instance memiliki antarmuka jaringan default (indeks 0) yang ditetapkan alamat IPv4 pribadi utama. Anda juga dapat menentukan alamat IPv4 privat tambahan, yang dikenal sebagai alamat IPv4 privat sekunder. Tidak seperti alamat IP privat primer, alamat IP privat sekunder dapat ditetapkan ulang dari satu instans ke instans lainnya. Untuk informasi selengkapnya, lihat Beberapa alamat IP.

Alamat IPv4 privat, terlepas dari apakah itu alamat primer atau sekunder, akan tetap dikaitkan dengan antarmuka jaringan saat instans dihentikan dan dimulai, atau dihibernasikan dan dimulai, dan dilepas saat instans diakhiri.

## Alamat IPv4 publik

Alamat IP publik adalah alamat IPv4 yang tidak dapat dijangkau dengan Internet. Anda dapat menggunakan alamat publik untuk komunikasi antara instans Anda dan Internet.

Saat Anda meluncurkan sebuah instans di VPC default, kami menetapkannya sebagai alamat IP publik secara default. Ketika Anda meluncurkan suatu instans ke dalam VPC non-default, subnet tersebut memiliki atribut yang menentukan apakah instans yang diluncurkan ke subnet tersebut menerima alamat IP publik dari kumpulan alamat IPv4 publik. Secara default, kami tidak menetapkan alamat IP publik ke instans yang diluncurkan di subnet non-default.

Anda dapat mengontrol apakah instans Anda menerima alamat IP publik sebagai berikut:

- Ubah atribut pengalamatan IP publik dari subnet Anda. Untuk informasi selengkapnya, lihat Memodifikasi atribut pengalamatan IPv4 publik untuk subnet Anda dalam Panduan Pengguna Amazon VPC.
- Aktifkan atau menonaktifkan fitur pengalamatan IP publik selama peluncuran. Ini menggantikan atribut pengalamatan IP publik subnet. Untuk informasi selengkapnya, lihat Tetapkan alamat IPv4 publik saat peluncuran.
- Batalkan penetapan alamat IP publik dari instans Anda setelah peluncuran. Untuk informasi selengkapnya, lihat the section called “Mengelola alamat IP”.

Alamat IP publik ditetapkan ke instans Anda dari kumpulan alamat IPv4 publik Amazon, dan tidak terkait dengan AWS akun Anda. Ketika alamat IP publik tidak dikaitkan dengan instans Anda, alamat IP tersebut dilepas kembali ke kumpulan IPv4 publik, dan Anda tidak dapat menggunakannya kembali.

Kami merilis alamat IP publik dari instance Anda dan menetapkan yang baru dalam kasus berikut:

- Kami merilis alamat IP publik ketika instance dihentikan, hibernasi, atau dihentikan. Kami menetapkan alamat IP publik baru saat Anda memulai instans yang dihentikan atau hibernasi.
- Kami merilis alamat IP publik saat Anda mengaitkan alamat IP Elastic dengan instance. Kami menetapkan alamat IP publik baru saat Anda memisahkan alamat IP Elastic dari instance Anda.
- Jika kami merilis alamat IP publik instans Anda dan memiliki antarmuka jaringan sekunder, kami tidak menetapkan alamat IP publik baru.
- Jika kami merilis alamat IP publik instans Anda dan memiliki alamat IP pribadi sekunder yang terkait dengan alamat IP Elastic, kami tidak menetapkan alamat IP publik baru.

Jika Anda memerlukan alamat IP publik yang persisten yang dapat dikaitkan dengan dan dari instans sesuai kebutuhan, gunakan alamat IP Elastis.

Jika Anda menggunakan DNS dinamis untuk memetakan nama DNS yang ada ke alamat IP publik instans baru, mungkin perlu waktu hingga 24 jam agar alamat IP tersebut tersebar melalui Internet. Akibatnya, instans baru mungkin tidak menerima traffic sedangkan instans yang dihentikan terus menerima permintaan. Untuk mengatasi masalah ini, gunakan alamat IP Elastis. Anda dapat mengalokasikan alamat IP Elastis Anda sendiri, dan mengaitkannya dengan instans Anda. Untuk informasi selengkapnya, lihat Alamat Elastic IP.

Jika Anda menggunakan Amazon VPC IP Address Manager (IPAM), Anda bisa mendapatkan blok alamat IPv4 publik yang berdekatan dari AWS dan menggunakannya untuk mengalokasikan alamat IP Elastic ke sumber daya. AWS Menggunakan blok alamat IPv4 yang berdekatan dapat secara signifikan mengurangi overhead manajemen untuk daftar kontrol akses keamanan dan menyederhanakan alokasi dan pelacakan alamat IP untuk penskalaan perusahaan. AWS Untuk informasi selengkapnya, lihat Meng alokasikan alamat IP Elastis berurutan dari kumpulan IPAM di Panduan Pengguna IPAM Amazon VPC.

Pertimbangan-pertimbangan

- AWS biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastic. Untuk informasi selengkapnya, lihat tab Alamat IPv4 Publik di halaman harga Amazon VPC.
- Instans yang mengakses instans lain melalui alamat IP NAT publiknya dikenai biaya untuk transfer data regional atau Internet, bergantung pada apakah instans tersebut berada di Wilayah yang sama.

## Pengoptimalan alamat IPv4 publik

AWS biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastic. Untuk informasi selengkapnya, lihat tab Alamat IPv4 Publik di halaman harga Amazon VPC.

Daftar berikut berisi tindakan yang dapat Anda lakukan untuk mengoptimalkan jumlah alamat IPv4 publik yang Anda gunakan:

- Gunakan penyeim bang beban elastis untuk menyeimbangkan lalu lintas ke instans EC2 Anda dan menonaktifkan IP Auto-assign publik pada ENI utama yang ditetapkan ke instans. Penyeimbang beban menggunakan satu alamat IPv4 publik, jadi ini mengurangi jumlah alamat IPv4 publik Anda. Anda mungkin juga ingin mengkonsolidasikan penyeimbang beban yang ada untuk lebih mengurangi jumlah alamat IPv4 publik.
- Jika satu-satunya alasan untuk menggunakan gateway NAT adalah untuk SSH ke instans EC2 di subnet pribadi untuk pemeliharaan atau keadaan darurat, pertimbangkan untuk menggunakan EC2 Instance Connect Endpoint sebagai gantinya. Dengan EC2 Instance Connect Endpoint, Anda dapat terhubung ke instans dari internet tanpa mengharuskan instance memiliki alamat IPv4 publik.
- Jika instans EC2 Anda berada di subnet publik dengan alamat IP publik yang dialokasikan untuknya, pertimbangkan untuk memindahkan instans ke subnet pribadi, menghapus alamat IP

publik, dan menggunakan gateway NAT publik untuk mengizinkan akses ke dan dari instans EC2 Anda. Ada pertimbangan biaya untuk menggunakan gateway NAT. Gunakan metode perhitungan ini untuk memutuskan apakah gateway NAT hemat biaya. Anda bisa mendapatkan yang `Number of public IPv4 addresses` diperlukan untuk perhitungan ini dengan membuat Laporan Biaya dan Penggunaan Pen AWS agihan.

```
NAT gateway per hour + NAT gateway public IPs + NAT gateway transfer / Existing
 public IP cost
```

Di mana:

- `NAT gateway per hour = $0.045 * 730 hours in a month * Number of Availability Zones the NAT gateways are in`
- `NAT gateway public IPs = $0.005 * 730 hours in a month * Number of IPs associated with your NAT gateways`
- `NAT gateway transfer = $0.045 * Number of GBs that will go through the NAT gateway in a month`
- `Existing public IP cost = $0.005 * 730 hours in a month * Number of public IPv4 addresses`

Jika totalnya kurang dari 1, gateway NAT lebih murah daripada alamat IPv4 publik.

- Gunakan AWS PrivateLink untuk terhubung secara pribadi ke AWS layanan atau layanan yang dihosting oleh AWS akun lain daripada menggunakan alamat IPv4 publik dan gateway internet.
- Bawa rentang alamat IP Anda sendiri (BYOIP) ke AWS dan gunakan rentang untuk alamat IPv4 publik daripada menggunakan alamat IPv4 Amazon-owned publik.
- Nonaktifkan alamat IPv4 publik yang ditetapkan secara otomatis untuk instans yang diluncurkan ke subnet. Opsi ini umumnya dinonaktifkan secara default untuk VPC saat Anda membuat subnet, tetapi Anda harus memeriksa subnet yang ada untuk memastikannya dinonaktifkan.
- Jika Anda memiliki instans EC2 yang tidak memerlukan alamat IPv4 publik, periksa apakah antarmuka jaringan yang terpasang pada instans Anda memiliki IP Auto-assign publik yang dinonaktifkan.
- Konfigurasikan titik akhir akselerator AWS Global Accelerator untuk instans EC2 di subnet pribadi untuk memungkinkan lalu lintas internet mengalir langsung ke titik akhir di VPC Anda tanpa memerlukan alamat IP publik. Anda juga dapat membawa alamat Anda sendiri ke AWS Global Accelerator dan menggunakan alamat IPv4 Anda sendiri untuk alamat IP statis akselerator Anda.

# Alamat IPv6

Alamat IPv6 bersifat unik secara global dan dapat dikonfigurasi agar tetap privat atau dapat dijangkau melalui Internet. Pengalamatan IPv6 publik dan pribadi tersedia di: AWS

- IPv6 pribadi: AWS mempertimbangkan alamat IPv6 pribadi yang tidak diiklankan dan tidak dapat diiklankan di Internet dari. AWS
- IPv6 Publik: AWS mempertimbangkan alamat IPv6 publik yang diiklankan di Internet dari. AWS

Untuk informasi selengkapnya tentang alamat IPv6 publik dan pribadi, lihat Alamat IPv6 di Panduan Pengguna Amazon VPC.

Semua jenis instans mendukung alamat IPv6 kecuali untuk yang berikut: C1, M1, M2, M3, dan T1.

Instans EC2 Anda menerima alamat IPv6 jika blok CIDR IPv6 dikaitkan dengan VPC dan subnet Anda, dan jika salah satu dari berikut ini benar:

- Subnet Anda dikonfigurasi untuk secara otomatis menetapkan alamat IPv6 ke sebuah instans selama peluncuran. Untuk informasi selengkapnya, lihat Memodifikasi atribut pengalamatan IP subnet Anda.
- Anda menetapkan alamat IPv6 ke instans Anda selama peluncuran.
- Anda menetapkan alamat IPv6 ke antarmuka jaringan primer instans Anda setelah peluncuran.
- Anda menetapkan alamat IPv6 ke antarmuka jaringan di subnet yang sama, dan menyertakan antarmuka jaringan ke instans Anda setelah peluncuran.

Ketika instance Anda menerima alamat IPv6 selama peluncuran, alamat tersebut dikaitkan dengan antarmuka jaringan utama (indeks 0) instans. Anda dapat mengelola alamat IPv6 untuk antarmuka jaringan utama instans Anda sebagai berikut:

- Tetapkan dan batalkan penetapan alamat IPv6 dari antarmuka jaringan. Jumlah alamat IPv6 yang dapat Anda tetapkan ke antarmuka jaringan dan jumlah antarmuka jaringan yang dapat Anda sertakan ke sebuah instans bervariasi tergantung tipe instans. Untuk informasi selengkapnya, lihat Alamat IP maksimum per antarmuka jaringan.
- Aktifkan alamat IPv6 primer. Alamat IPv6 primer memungkinkan Anda untuk menghindari mengganggu lalu lintas ke instans atau ENI. Untuk informasi lebih lanjut, lihat Membuat antarmuka jaringan untuk instans EC2 Anda atau Kelola alamat IP untuk antarmuka jaringan Anda.

Alamat IPv6 tetap ada saat Anda menghentikan dan memulai, atau menghibernasikan dan memulai instans Anda, dan akan dilepas saat Anda menghentikan instans. Anda tidak dapat menetapkan ulang alamat IPv6 saat ditetapkan ke antarmuka jaringan lain—Anda harus membatalkan penetapannya terlebih dahulu.

Anda dapat mengontrol apakah instance dapat dijangkau melalui alamat IPv6-nya dengan mengontrol routing untuk subnet atau dengan menggunakan grup keamanan dan aturan ACL jaringan. Untuk informasi selengkapnya, lihat Privasi lalu lintas Internetwork di Panduan Pengguna Amazon VPC.

Untuk informasi selengkapnya tentang rentang alamat IPv6 yang dicadangkan, lihat Registri Alamat IPv6 I ANA dan Special-Purpose RFC4291.

## Beberapa alamat IP

Anda dapat menentukan beberapa alamat IPv4 dan IPv6 privat untuk instans Anda. Jumlah antarmuka jaringan dan alamat IPv4 dan IPv6 privat yang dapat Anda tentukan untuk sebuah instans bergantung pada tipe instans. Untuk informasi selengkapnya, lihat Alamat IP maksimum per antarmuka jaringan.

Kasus penggunaan

- Melakukan hosting beberapa situs web di satu server dengan menggunakan beberapa sertifikat SSL di satu server dan mengaitkan setiap sertifikat dengan alamat IP tertentu.
- Mengoperasikan peralatan jaringan, seperti firewall atau load balancer, yang memiliki beberapa alamat IP untuk setiap antarmuka jaringan.
- Mengalihkan traffic internal ke instans siaga jika instans Anda gagal, dengan menetapkan kembali alamat IP sekunder ke instans siaga.

Cara kerja beberapa alamat IP

- Anda dapat menetapkan alamat IPv4 privat sekunder ke antarmuka jaringan apa pun.
- Anda dapat menetapkan beberapa alamat IPv6 ke antarmuka jaringan yang ada di subnet yang memiliki blok CIDR IPv6 terkait.
- Anda harus memilih alamat IPv4 sekunder dari rentang blok CIDR IPv4 subnet untuk antarmuka jaringan.
- Anda harus memilih alamat IPv6 dari rentang blok CIDR IPv6 subnet untuk antarmuka jaringan.

- Anda mengaitkan grup keamanan dengan antarmuka jaringan, bukan alamat IP individu. Oleh karena itu, setiap alamat IP yang Anda tentukan dalam antarmuka jaringan tunduk pada grup keamanan antarmuka jaringannya.
- Beberapa alamat IP dapat ditetapkan dan tidak ditetapkan ke antarmuka jaringan yang disertakan ke instans yang berjalan atau dihentikan.
- Alamat IPv4 privat sekunder yang ditetapkan ke antarmuka jaringan dapat ditetapkan kembali ke antarmuka jaringan lain jika Anda mengizinkannya secara eksplisit.
- Alamat IPv6 tidak dapat dialihkan ke antarmuka jaringan lain; Anda harus terlebih dahulu membatalkan penetapan alamat IPv6 dari antarmuka jaringan yang ada.
- Saat menetapkan beberapa alamat IP ke antarmuka jaringan menggunakan alat baris perintah atau API, seluruh operasi gagal jika salah satu alamat IP tidak dapat ditetapkan.
- Alamat IPv4 privat primer, alamat IPv4 privat sekunder, alamat IP Elastis, dan alamat IPv6 tetap menggunakan antarmuka jaringan sekunder saat dilepaskan dari instans atau disertakan ke instans.
- Meskipun Anda tidak dapat melepaskan antarmuka jaringan primer dari sebuah instans, Anda dapat menetapkan kembali alamat IPv4 privat sekunder dari antarmuka jaringan primer ke antarmuka jaringan lain.

Untuk informasi selengkapnya, lihat the section called “Alamat IP sekunder”.

## Nama host instans EC2

Saat Anda membuat instance EC2, buat AWS nama host untuk instance tersebut. Untuk informasi selengkapnya tentang jenis nama host dan cara penyediaannya AWS, lihat. Nama host dan domain instans EC2 Amazon menyediakan server DNS yang menyelesaikan Amazon-provided nama host ke alamat IPv4 dan IPv6. Server Amazon DNS terletak di dasar rentang jaringan VPC Anda plus dua. Untuk informasi selengkapnya, lihat Atribut DNS untuk VPC Anda di Panduan Pengguna Amazon VPC.

## Link-local alamat

Amazon EC2 menyimpan alamat link-local IPv4 tertentu untuk layanan yang hanya dapat diakses dari instans EC2. Layanan ini berjalan pada host yang mendasarinya, bukan pada instance. Saat Anda mengakses alamat link-local yang dicadangkan untuk layanan ini, Anda berkomunikasi dengan Xen hypervisor atau Nitro controller. Host yang mendasarinya menangani lalu lintas ke alamat yang

dicadangkan ini alih-alih meneruskannya sesuai dengan rute yang dikonfigurasi pelanggan. Grup keamanan dan ACL jaringan tidak memfilter lalu lintas ini.

AWS tidak menyimpan setiap alamat IPv4 di`169.254.0.0/16`. Lalu lintas ke alamat non-reservasi dalam rentang ini mengikuti rute VPC yang berlaku. Grup keamanan dan ACL jaringan Anda dapat memfilter lalu lintas ini.

Link-local rentang alamat

- IPv4 — 169.254.0. 0/16 (169.254.0.0 hingga 169.254.255.255)
- IPv6 – fe80::/10

Layanan yang Anda akses menggunakan alamat link-lokal

- Layanan Metadata Instance - `169.254.169.254`
- Amazon Route 53 Resolver(juga dikenal sebagai server DNS Amazon) - `169.254.169.253`
- Layanan Sinkronisasi Waktu Amazon - `169.254.169.123`
- AWS Server KMS

## Mengelola alamat IPv4 untuk instans EC2 Anda

Anda dapat menetapkan alamat IPv4 publik ke instans Anda saat meluncurkannya. Anda dapat melihat alamat IPv4 untuk instans Anda di konsol melalui halaman Instans atau halaman Antarmuka Jaringan.

Tugas

- Tetapkan alamat IPv4 publik saat peluncuran
- Tetapkan alamat IPv4 pribadi saat peluncuran
- Lihat alamat IPv4 utama
- Melihat alamat IPv4 menggunakan metadata instance

### Tetapkan alamat IPv4 publik saat peluncuran

Setiap subnet memiliki atribut yang menentukan apakah instans yang diluncurkan ke subnet tersebut diberi alamat IP publik. Secara default, subnet non-default mengatur atribut ini ke false, dan subnet default mengatur atribut ini ke true. Saat Anda meluncurkan instans, fitur pengalamatan IPv4 publik

juga tersedia bagi Anda untuk mengontrol apakah instans Anda diberi alamat IPv4 publik; Anda dapat mengganti perilaku default atribut pengalamatan IP subnet. Alamat IPv4 publik ditetapkan dari kumpulan alamat IPv4 publik Amazon, dan ditetapkan ke antarmuka jaringan dengan indeks perangkat 0. Fitur ini bergantung pada kondisi tertentu pada saat Anda meluncurkan instans Anda.

Pertimbangan-pertimbangan

- Anda dapat membatalkan penetapan alamat IP publik dari instance Anda setelah diluncurkan dengan mengelola alamat IP yang terkait dengan antarmuka jaringan. Untuk informasi selengkapnya tentang alamat IPv4 publik, lihat Alamat IPv4 publik.
- Anda tidak dapat menetapkan alamat IP publik secara otomatis jika Anda menentukan lebih dari satu antarmuka jaringan. Selain itu, Anda tidak dapat mengganti setelan subnet menggunakan fitur IP publik menetapkan otomatis jika Anda menentukan antarmuka jaringan yang ada untuk indeks perangkat 0.
- Apakah Anda menetapkan alamat IP publik ke instans selama peluncuran atau tidak, Anda dapat mengaitkan alamat IP Elastic dengan instance setelah diluncurkan. Untuk informasi selengkapnya, lihat Alamat Elastic IP. Anda juga dapat mengubah perilaku pengalamatan IPv4 publik subnet Anda. Untuk informasi selengkapnya, lihat Memodifikasi atribut pengalamatan IPv4 publik untuk subnet Anda.

Console

Untuk menetapkan alamat IPv4 publik saat peluncuran

Ikuti prosedur untuk meluncurkan instance, dan ketika Anda mengkonfigurasi Peng aturan Jaringan, pilih opsi untuk IP Auto-assign Publik.

AWS CLI

Untuk menetapkan alamat IPv4 publik saat peluncuran

Gunakan perintah run-instance dengan opsi. `--associate-public-ip-address`

```
--associate-public-ip-address
```

PowerShell

Untuk menetapkan alamat IPv4 publik saat peluncuran

Gunakan New-EC2Instance cmdlet dengan parameter. `-AssociatePublicIp`

```
-AssociatePublicIp $true
```

## Tetapkan alamat IPv4 pribadi saat peluncuran

Anda dapat menentukan alamat IPv4 pribadi dari rentang alamat IPv4 subnet, atau membiarkan Amazon EC2 memilihnya untuk Anda. Alamat ini ditetapkan ke antarmuka jaringan utama.

Untuk menetapkan alamat IPv4 setelah peluncuran, lihatthe section called “Menetapkan alamat IP sekunder ke sebuah instance”.

Console

Untuk menetapkan alamat IPv4 pribadi saat peluncuran

Ikuti prosedur untuk meluncurkan instans. Saat Anda mengkonfigurasi Pengaturan Jaringan, per luas Konfigurasi jaringan lanjutan dan masukkan nilai untuk IP Utama.

AWS CLI

Untuk menetapkan alamat IPv4 pribadi saat peluncuran

Gunakan perintah run-instance dengan opsi. `--private-ip-address`

```
--private-ip-addresses 10.251.50.12
```

Untuk membiarkan Amazon EC2 memilih alamat IP, hilangkan opsi ini.

PowerShell

Untuk menetapkan alamat IPv4 pribadi saat peluncuran

Gunakan New-EC2Instance cmdlet dengan parameter. `-PrivateIpAddress`

```
-PrivateIpAddress 10.251.50.12
```

Untuk membiarkan Amazon EC2 memilih alamat IP, hilangkan parameter ini.

## Lihat alamat IPv4 utama

Alamat IPv4 publik ditampilkan sebagai properti antarmuka jaringan di konsol, tetapi dipetakan ke alamat IPv4 privat primer melalui NAT. Oleh karena itu, jika Anda memeriksa properti antarmuka

jaringan pada instans Anda, misalnya melalui `ifconfig` (Linux) atau `ipconfig` (Windows), alamat IPv4 publik tidak ditampilkan.

Console

Untuk melihat alamat IPv4 untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pada tab Jaringan, cari Alamat IPv4 Publik dan Alamat IPv4 Pribadi.
5. (Opsional) Tab Jaringan juga berisi antarmuka jaringan dan alamat IP elastis untuk instance.

AWS CLI

Untuk melihat alamat IPv4 utama untuk sebuah instance

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query "Reservations[].Instances[].PrivateIpAddress" \
    --output text
```

Berikut ini adalah output contoh.

```
10.251.50.12
```

PowerShell

Untuk melihat alamat IPv4 utama untuk sebuah instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances.PrivateIpAddress
```

Berikut ini adalah output contoh.

```
10.251.50.12
```

## Melihat alamat IPv4 menggunakan metadata instance

Anda bisa mendapatkan alamat IPv4 untuk instance Anda dengan mengambil metadata instans. Untuk informasi selengkapnya, lihat Menggunakan metadata instans untuk mengelola instans EC2 Anda.

Untuk melihat alamat IPv4 menggunakan metadata instance

1. Connect ke instans Anda. Untuk informasi selengkapnya, lihat Sambungkan ke instans EC2.
2. Gunakan salah satu perintah berikut ini.

   IMDSv2

   Linux

   Jalankan perintah berikut dari instance Linux Anda.

   ```
   TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
   metadata-token-ttl-seconds: 21600"` \
   && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
   meta-data/local-ipv4
   ```

   Windows

   Jalankan perintah berikut dari instance Windows Anda.

   ```
   [string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
   seconds" = "21600"} `
       -Method PUT -Uri http://169.254.169.254/latest/api/token
   ```

   ```
   Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
       -Method GET -Uri http://169.254.169.254/latest/meta-data/local-ipv4
   ```

   IMDSv1

   Linux

   Jalankan perintah berikut dari instance Linux Anda.

   ```
   curl http://169.254.169.254/latest/meta-data/local-ipv4
   ```

Windows

Jalankan perintah berikut dari instance Windows Anda.

```
Invoke-RestMethod http://169.254.169.254/latest/meta-data/local-ipv4
```

3. Gunakan salah satu perintah berikut untuk mengakses alamat IP publik. Jika ada alamat IP Elastic yang terkait dengan instance, perintah mengembalikan alamat IP Elastic.

IMDSv2

Linux

Jalankan perintah berikut dari instance Linux Anda.

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
 "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/public-ipv4
```

Windows

Jalankan perintah berikut dari instance Windows Anda.

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-
seconds" = "21600"} `
    -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
    -Method GET -Uri http://169.254.169.254/latest/meta-data/public-ipv4
```

IMDSv1

Linux

Jalankan perintah berikut dari instance Linux Anda.

```
curl http://169.254.169.254/latest/meta-data/public-ipv4
```

Windows

Jalankan perintah berikut dari instance Windows Anda.

```
Invoke-RestMethod http://169.254.169.254/latest/meta-data/public-ipv4
```

## Mengelola alamat IPv6 untuk instans EC2 Anda

Jika VPC dan subnet Anda memiliki blok CIDR IPv6 yang dikaitkan dengannya, Anda dapat menetapkan alamat IPv6 ke instans Anda selama atau setelah peluncuran. Anda dapat mengakses alamat IPv6 untuk instans Anda di konsol di halaman Instans atau halaman Antar muka Jaringan. Tugas berikut mengkonfigurasi alamat IP untuk instans Anda. Untuk mengonfigurasi alamat IP untuk antarmuka jaringan Anda sebagai gantinya, lihatthe section called “Mengelola alamat IP”.

Tugas

- Menetapkan alamat IPv6 ke sebuah instans
- Melihat alamat IPv6 untuk sebuah instance
- Melihat alamat IPv6 menggunakan metadata instance
- Untuk membatalkan penetapan alamat IPv6 dari sebuah instans

### Menetapkan alamat IPv6 ke sebuah instans

Anda dapat menentukan alamat IPv6 dari rentang alamat IPv6 subnet, atau membiarkan Amazon EC2 memilihnya untuk Anda. Alamat ini ditetapkan ke antarmuka jaringan utama. Perhatikan bahwa jenis instance berikut tidak mendukung alamat IPv6: C1, M1, M2, M3, dan T1.

Console

Untuk menetapkan alamat IPv6 saat peluncuran

Ikuti prosedur untuk meluncurkan instans. Saat Anda mengkonfigurasi Pengaturan Jaringan, pilih opsi untuk Auto-assign IPv6 IP. Jika Anda tidak melihat opsi ini, subnet yang dipilih tidak memiliki blok IPv6 CIDR terkait.

Untuk menetapkan alamat IPv6 setelah peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.

3. Pilih instans Anda, dan pilih Tindakan, Jaringan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan. Di alamat IPv6, pilih Tetapkan alamat IP baru.
5. Masukkan alamat IPv6 dari rentang subnet, atau biarkan bidang kosong untuk membiarkan Amazon EC2 memilih alamat IPv6 untuk Anda. Jika Anda tidak melihat opsi ini, subnet instance tidak memiliki blok CIDR IPv6 terkait.
6. Pilih Simpan.

AWS CLI

Untuk menetapkan alamat IPv6 saat peluncuran

Gunakan perintah run-instance dengan opsi. `--ipv6-addresses` Contoh berikut menetapkan dua alamat IPv6.

```
--ipv6-addresses Ipv6Address=2001:db8::1234:5678:1.2.3.4
 Ipv6Address=2001:db8::1234:5678:5.6.7.8
```

Untuk membiarkan Amazon EC2 memilih alamat IPv6, gunakan `--ipv6-address-count` opsi sebagai gantinya. Contoh berikut menetapkan dua alamat IPv6.

```
--ipv6-address-count 2
```

Untuk menetapkan alamat IPv6 setelah peluncuran

Gunakan perintah attach -ipv6-address. Contoh berikut menetapkan dua alamat IPv6.

```
aws ec2 assign-ipv6-addresses \
    --network-interface-id eni-1234567890abcdef0 \
    --ipv6-addresses 2001:db8::1234:5678:1.2.3.4 2001:db8::1234:5678:5.6.7.8
```

Untuk membiarkan Amazon EC2 memilih alamat IPv6, gunakan `--ipv6-address-count` opsi sebagai gantinya. Contoh berikut menetapkan dua alamat IPv6.

```
aws ec2 assign-ipv6-addresses \
    --network-interface-id eni-1234567890abcdef0 \
    --ipv6-address-count 2
```

PowerShell

Untuk menetapkan alamat IPv6 saat peluncuran

Gunakan New-EC2Instance cmdlet dengan parameter. `-Ipv6Address` Contoh berikut menetapkan dua alamat IPv6.

```
-Ipv6Address $ipv6addr1,$ipv6addr2
```

Tentukan alamat IPv6 sebagai berikut.

```
$ipv6addr1 = New-Object Amazon.EC2.Model.InstanceIpv6Address
$ipv6addr1.Ipv6Address = "2001:db8::1234:5678:1.2.3.4"
$ipv6addr2 = New-Object Amazon.EC2.Model.InstanceIpv6Address
$ipv6addr2.Ipv6Address = "2001:db8::1234:5678:5.6.7.8"
```

Untuk membiarkan Amazon EC2 memilih alamat IPv6, gunakan `-Ipv6AddressCount` parameter sebagai gantinya. Contoh berikut menetapkan dua alamat IPv6.

```
-Ipv6AddressCount 2
```

Untuk menetapkan alamat IPv6 setelah peluncuran

Gunakan Register-EC2Ipv6AddressList cmdlet. Contoh berikut menetapkan dua alamat IPv6.

```
Register-EC2Ipv6AddressList `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Ipv6Address "2001:db8::1234:5678:1.2.3.4","2001:db8::1234:5678:5.6.7.8"
```

Untuk membiarkan Amazon EC2 memilih alamat IPv6, gunakan `-Ipv6AddressCount` parameter sebagai gantinya. Contoh berikut menetapkan dua alamat IPv6.

```
Register-EC2Ipv6AddressList `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Ipv6AddressCount 2
```

## Melihat alamat IPv6 untuk sebuah instance

Anda dapat melihat alamat IPv6 untuk instans Anda.

Console

Untuk melihat alamat IPv6 untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pada tab Jaringan, tempatkan alamat IPv6.

AWS CLI

Untuk melihat alamat IPv6 untuk sebuah instance

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query "Reservations[*].Instances[].Ipv6Address" \
    --output text
```

Berikut ini adalah output contoh.

```
2001:db8::1234:5678:1.2.3.4
```

PowerShell

Untuk melihat alamat IPv6 untuk sebuah instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances.Ipv6Address
```

Berikut ini adalah output contoh.

```
2001:db8::1234:5678:1.2.3.4
```

## Melihat alamat IPv6 menggunakan metadata instance

Setelah terhubung ke instans, Anda dapat mengambil alamat IPv6 menggunakan metadata instans. Pertama, Anda harus mendapatkan alamat MAC instance dari`http://169.254.169.254/latest/meta-data/network/interfaces/macs/`.

### IMDSv2

#### Linux

Jalankan perintah berikut dari instance Linux Anda.

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

#### Windows

Jalankan cmdlet berikut dari instance Windows Anda.

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
    -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} `
    -Method GET -Uri http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

### IMDSv1

#### Linux

Jalankan perintah berikut dari instance Linux Anda.

```
curl http://169.254.169.254/latest/meta-data/network/interfaces/macs/mac-address/ipv6s
```

#### Windows

Jalankan cmdlet berikut dari instance Windows Anda.

```
Invoke-RestMethod -Uri http://169.254.169.254/latest/meta-data/network/interfaces/
macs/mac-address/ipv6s
```

## Untuk membatalkan penetapan alamat IPv6 dari sebuah instans

Anda dapat membatalkan penetapan alamat IPv6 dari instans kapan saja.

### Console

Untuk membatalkan penetapan alamat IPv6 dari sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda, dan pilih Tindakan, Jaringan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan. Di bawah alamat IPv6, pilih Batalkan penetapan di samping alamat IPv6.
5. Pilih Simpan.

### AWS CLI

Untuk membatalkan penetapan alamat IPv6 dari sebuah instans

Gunakan perintah unattach-ipv6-address.

```
aws ec2 unassign-ipv6-addresses \
    --network-interface-id eni-1234567890abcdef0 \
    --ipv6-addresses 2001:db8::1234:5678:1.2.3.4
```

### PowerShell

Untuk membatalkan penetapan alamat IPv6 dari sebuah instans

Gunakan Unregister-EC2Ipv6AddressList cmdlet.

```
Unregister-EC2Ipv6AddressList `
```

```
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Ipv6Address 2001:db8::1234:5678:1.2.3.4
```

# Alamat IP sekunder untuk instans EC2

Alamat IPv4 pertama yang ditetapkan ke antarmuka jaringan dikenal sebagai alamat IP primer. Alamat IP sekunder adalah alamat IPv4 tambahan yang ditetapkan ke antarmuka jaringan. Untuk informasi selengkapnya, lihat the section called “Beberapa alamat IP”.

Anda juga dapat menetapkan beberapa alamat IPv6 ke sebuah instance. Untuk informasi selengkapnya, lihat the section called “Alamat IPv6”.

Tugas

- Menetapkan alamat IP sekunder ke sebuah instance
- Konfigurasikan sistem operasi untuk menggunakan alamat IP sekunder
- Membatalkan penetapan alamat IP sekunder dari sebuah instance

## Menetapkan alamat IP sekunder ke sebuah instance

Anda dapat menetapkan alamat IP sekunder ke antarmuka jaringan untuk sebuah instance saat Anda meluncurkan instans, atau setelah instance berjalan.

Console

Untuk menetapkan alamat IP sekunder saat peluncuran

1. Ikuti prosedur untuk meluncurkan instans. Saat Anda mengkonfigurasi Peng aturan Jaringan, perluas Konfigurasi jaringan lanjutan.
2. Untuk IP Sekunder, pilih Tetapkan secara otomatis dan masukkan jumlah alamat IP yang akan ditetapkan oleh Amazon EC2. Atau, pilih Tet apkan secara manual dan masukkan alamat IPv4.
3. Selesaikan langkah-langkah selanjutnya untuk meluncurkan instans.

Untuk menetapkan alamat IP sekunder setelah peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda, dan pilih Tindakan, Jaringan, Kelola alamat IP.
4. Bentangkan antarmuka jaringan.
5. Untuk menambahkan alamat IPv4, di bawah alamat IPv4, pilih Tet apkan alamat IP baru. Masukkan alamat IPv4 dari rentang subnet, atau biarkan bidang kosong untuk membiarkan Amazon EC2 memilihnya untuk Anda.
6. Pilih Simpan.

AWS CLI

Untuk menetapkan alamat IP sekunder saat peluncuran

Gunakan perintah run-instance dengan opsi. `--secondary-private-ip-addresses`

```
--secondary-private-ip-addresses 10.251.50.12
```

Untuk membiarkan Amazon EC2 memilih alamat IP, gunakan `--secondary-private-ip-address-count` opsi sebagai gantinya. Contoh berikut menetapkan satu alamat IP sekunder.

```
--secondary-private-ip-address-count 1
```

Atau, Anda dapat membuat antarmuka jaringan. Untuk informasi selengkapnya, lihat the section called “Membuat antarmuka jaringan”.

Untuk menetapkan alamat IP sekunder setelah peluncuran

Gunakan perintah attach -private-ip-address dengan opsi. `--private-ip-addresses`

```
aws ec2 assign-private-ip-addresses \
    --network-interface-ids eni-1234567890abcdef0 \
    --private-ip-addresses 10.251.50.12
```

Untuk membiarkan Amazon EC2 memilih alamat IPv4, gunakan `--secondary-private-ip-address-count` parameter sebagai gantinya. Contoh berikut menetapkan satu alamat IPv4.

```
aws ec2 assign-private-ip-addresses \
    --network-interface-ids eni-1234567890abcdef0 \
```

```
    --secondary-private-ip-address-count 1
```

PowerShell

Untuk menetapkan alamat IP sekunder saat peluncuran

Anda harus membuat antarmuka jaringan. Untuk informasi selengkapnya, lihat the section called "Membuat antarmuka jaringan".

Untuk menetapkan alamat IP sekunder setelah peluncuran

Gunakan Register-EC2PrivateIpAddress cmdlet dengan parameter. `-PrivateIpAddress`

```
Register-EC2PrivateIpAddress `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -PrivateIpAddress 10.251.50.12
```

Untuk membiarkan Amazon EC2 memilih alamat IPv4, gunakan `-SecondaryPrivateIpAddressCount` parameter sebagai gantinya. Contoh berikut menetapkan satu alamat IPv4.

```
Register-EC2PrivateIpAddress `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -SecondaryPrivateIpAddressCount 1
```

## Konfigurasikan sistem operasi untuk menggunakan alamat IP sekunder

Setelah menetapkan alamat IP sekunder ke instans, Anda harus mengonfigurasi sistem operasi pada instans Anda untuk mengenali alamat IPv4 pribadi tambahan.

Instans Linux

- Jika Anda menggunakan Amazon Linux, paket ec2-net-utils dapat menangani langkah ini untuk Anda. Paket tersebut mengonfigurasi antarmuka jaringan tambahan yang Anda sertakan saat instans berjalan, merefresh alamat IPv4 sekunder selama pembaruan DHCP lease, dan memperbarui aturan perutean terkait. Anda dapat segera menyegarkan daftar antarmuka dengan menggunakan salah satu perintah berikut, tergantung pada sistem Anda: `sudo systemctl restart systemd-networkd` (AL2023) atau `sudo service network restart` (Amazon Linux 2). Anda dapat melihat daftar terbaru menggunakan perintah berikut:`ip addr li`. Jika Anda

memerlukan kontrol manual atas konfigurasi jaringan Anda, Anda dapat menghapus paket ec2-net-utils. Untuk informasi selengkapnya, lihat Meng onfigurasi antarmuka jaringan Anda menggunakan ec2- net-utils.

- Jika Anda menggunakan distribusi Linux lain, lihat dokumentasi untuk distribusi Linux Anda. Cari informasi tentang cara mengonfigurasi antarmuka jaringan dan alamat IPv4 sekunder tambahan. Jika instans memiliki dua atau beberapa antarmuka di subnet yang sama, cari informasi tentang menggunakan aturan perutean untuk mengatasi perutean asimetris.

Instans Windows

Untuk informasi selengkapnya, lihat Konfigurasikan alamat IPv4 pribadi sekunder untuk instans Windows.

## Membatalkan penetapan alamat IP sekunder dari sebuah instance

Jika Anda tidak lagi memerlukan alamat IP sekunder, Anda dapat membatalkan tetapannya dari instance atau antarmuka jaringan. Ketika alamat IPv4 privat sekunder tidak ditetapkan dari antarmuka jaringan, alamat IP Elastis (jika ada) juga akan dipisahkan.

Console

Untuk membatalkan penetapan alamat IPv4 privat sekunder dari sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih Tindakan, Jaringan, Kelola Alamat IP.
4. Bentangkan antarmuka jaringan. Untuk alamat IPv4, pilih Batalkan penetapan untuk alamat IPv4 yang akan dibatalkan penetapan.
5. Pilih Simpan.

AWS CLI

Untuk membatalkan penetapan alamat IP pribadi sekunder

Gunakan perintah unattach-private-ip-address.

```
aws ec2 unassign-private-ip-addresses \
```

```
    --network-interface eni-1234567890abcdef0\
    --private-ip-addresses 10.251.50.12
```

PowerShell

Untuk membatalkan penetapan alamat IP pribadi sekunder

Gunakan Unregister-EC2PrivateIpAddress cmdlet.

```
Unregister-EC2PrivateIpAddress `
    -NetworkInterface eni-1234567890abcdef0 `
    -PrivateIpAddress 10.251.50.12
```

## Konfigurasikan alamat IPv4 pribadi sekunder untuk instans Windows

Anda dapat menentukan banyak alamat IPv4 privat untuk instans Anda. Setelah Anda menetapkan alamat IPv4 privat sekunder ke sebuah instans, Anda harus mengonfigurasi sistem operasi pada instans untuk mengenali alamat IPv4 privat sekunder.

> **Note**
>
> Instruksi ini didasarkan pada Windows Server 2022. Implementasi langkah-langkah ini mungkin bervariasi berdasarkan sistem operasi instance Windows.

Tugas

- Prasyarat
- Langkah 1: Konfigurasikan pengalamatan IP statis di instans Anda
- Langkah 2: Konfigurasikan alamat IP privat sekunder untuk instans Anda
- Langkah 3: Konfigurasikan aplikasi untuk Menggunakan alamat IP privat sekunder

### Prasyarat

- Tetapkan alamat IPv4 privat sekunder ke antarmuka jaringan untuk instans. Anda dapat menetapkan alamat IPv4 privat sekunder saat Anda meluncurkan instans, atau setelah instans berjalan. Untuk informasi selengkapnya, lihat Menetapkan alamat IP sekunder ke sebuah instance.

## Langkah 1: Konfigurasikan pengalamatan IP statis di instans Anda

Untuk mengaktifkan instans Windows Anda untuk menggunakan banyak alamat IP, Anda harus mengonfigurasi instans Anda untuk menggunakan pengalamatan IP statis, bukan server DHCP.

> **Important**
>
> Saat Anda mengonfigurasi pengalamatan IP statis di instance Anda, alamat IP harus sesuai dengan apa yang ditampilkan di konsol, CLI, atau API. Jika Anda salah memasukkan alamat IP ini, instans bisa jadi tidak dapat dijangkau.

Untuk mengonfigurasi pengalamatan IP statis pada instans Windows

1. Hubungkan ke instans Anda.
2. Temukan alamat IP, subnet mask, dan alamat gateway default untuk instans dengan melakukan langkah-langkah berikut:
   - Jalankan perintah berikut di PowerShell:

     ```
     ipconfig /all
     ```

     Tinjau output dan catat nilai Alamat IPv4, Subnet Mask, Default Gateway, dan Server DNS untuk antarmuka jaringan. Output Anda harus menyerupai contoh berikut:

     ```
     ...

     Ethernet adapter Ethernet 4:

        Connection-specific DNS Suffix  . : us-west-2.compute.internal
        Description . . . . . . . . . . . : Amazon Elastic Network Adapter #2
        Physical Address. . . . . . . . . : 02-9C-3B-FC-8E-67
        DHCP Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        Link-local IPv6 Address . . . . . : fe80::f4d1:a773:5afa:cd1%7(Preferred)
        IPv4 Address. . . . . . . . . . . : 10.200.0.128(Preferred)
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Lease Obtained. . . . . . . . . . : Monday, April 8, 2024 12:19:29 PM
        Lease Expires . . . . . . . . . . : Monday, April 8, 2024 4:49:30 PM
        Default Gateway . . . . . . . . . : 10.200.0.1
        DHCP Server . . . . . . . . . . . : 10.200.0.1
     ```

```
   DHCPv6 IAID . . . . . . . . . . . : 151166011
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2D-67-AC-FC-12-34-9A-BE-A5-
E7
   DNS Servers . . . . . . . . . . . : 10.200.0.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

3. Buka Pusat Jaringan dan Berbagi dengan menjalankan perintah berikut di PowerShell:

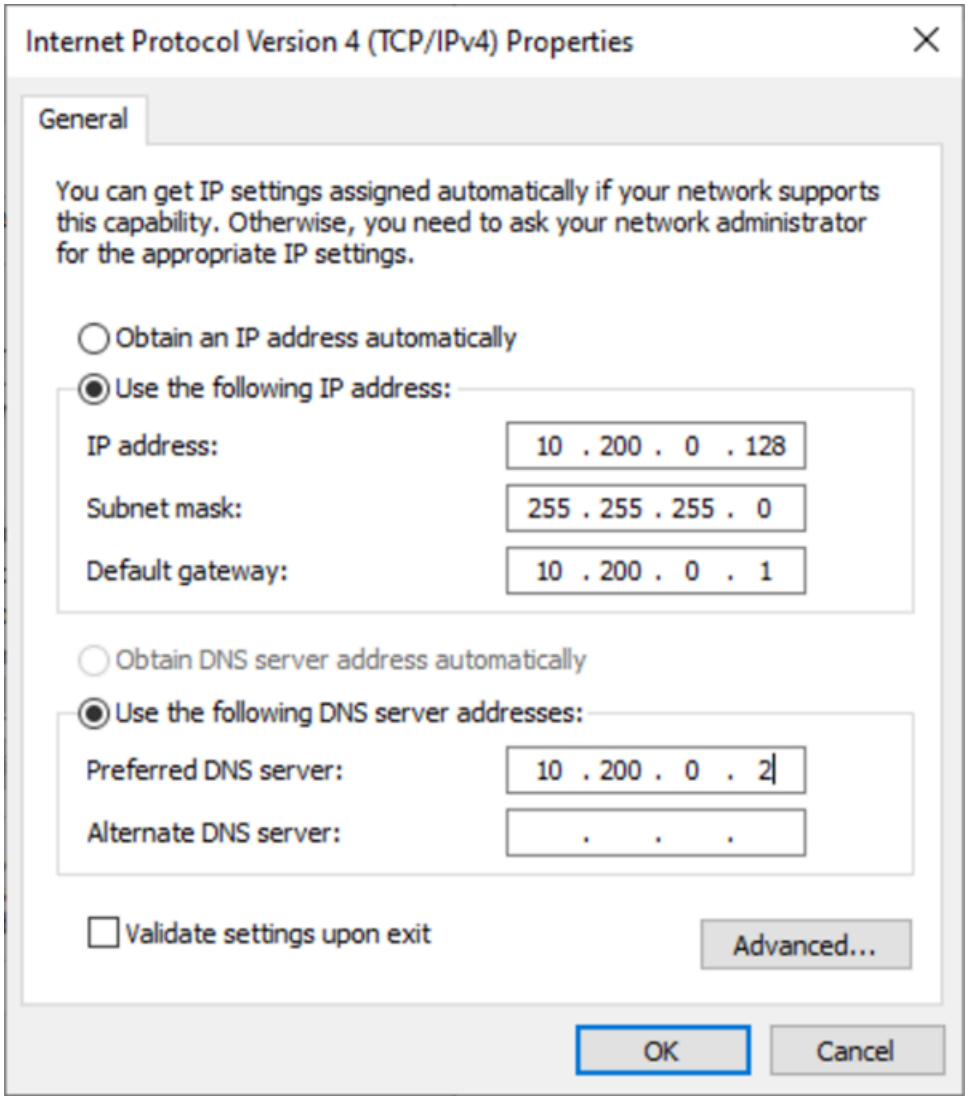
```
& $env:SystemRoot\system32\control.exe ncpa.cpl
```

4. Buka menu konteks (klik kanan) untuk antarmuka jaringan (Local Area Connection atau Ethernet) dan pilih Properties.
5. Pilih Protokol Internet Versi 4 (TCP/IPv4), Pro perti.
6. Dalam kotak dialog Properti Protokol Internet Versi 4 (TCP/IPv4), pilih Gunakan alamat IP berikut, masukkan nilai berikut, lalu pilih OK.

| Bidang | Nilai |
|---|---|
| Alamat IP | Alamat IPv4 diperoleh pada langkah 2 di atas. |
| Subnet mask | Subnet mask diperoleh pada langkah 2 di atas. |
| Gateway default | Alamat gateway default diperoleh pada langkah 2 di atas. |
| Server DNS pilihan | Server DNS diperoleh pada langkah 2 di atas. |
| Server DNS alternatif | Server DNS alternatif diperoleh pada langkah 2 di atas. Jika server DNS alternatif tidak terdaftar, biarkan bidang ini kosong. |

**Important**

Jika Anda menyetel alamat IP ke nilai apa pun selain alamat IP saat ini, Anda akan kehilangan konektivitas ke instans.



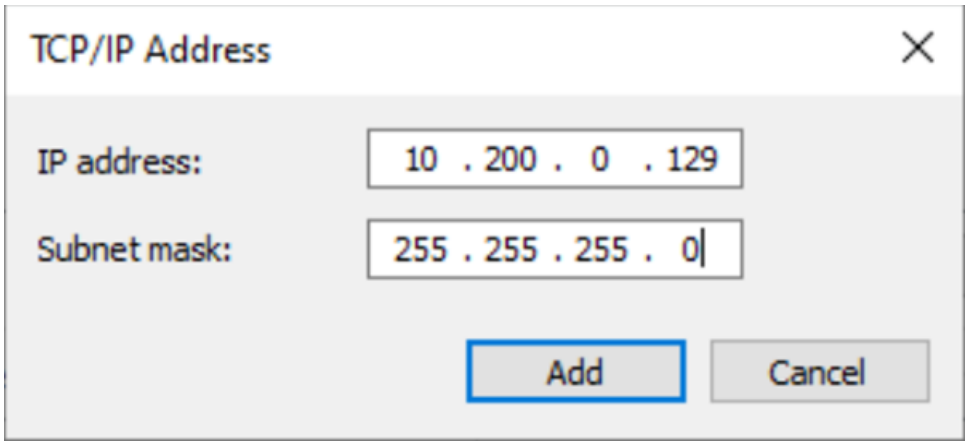

Anda akan kehilangan konektivitas RDP ke instans Windows selama beberapa detik saat instans tersebut diubah dari menggunakan DHCP menjadi pengalamatan statis. Instans tersebut mempertahankan informasi alamat IP yang sama seperti sebelumnya, tetapi sekarang informasi ini statis dan tidak dikelola oleh DHCP.

## Langkah 2: Konfigurasikan alamat IP privat sekunder untuk instans Anda

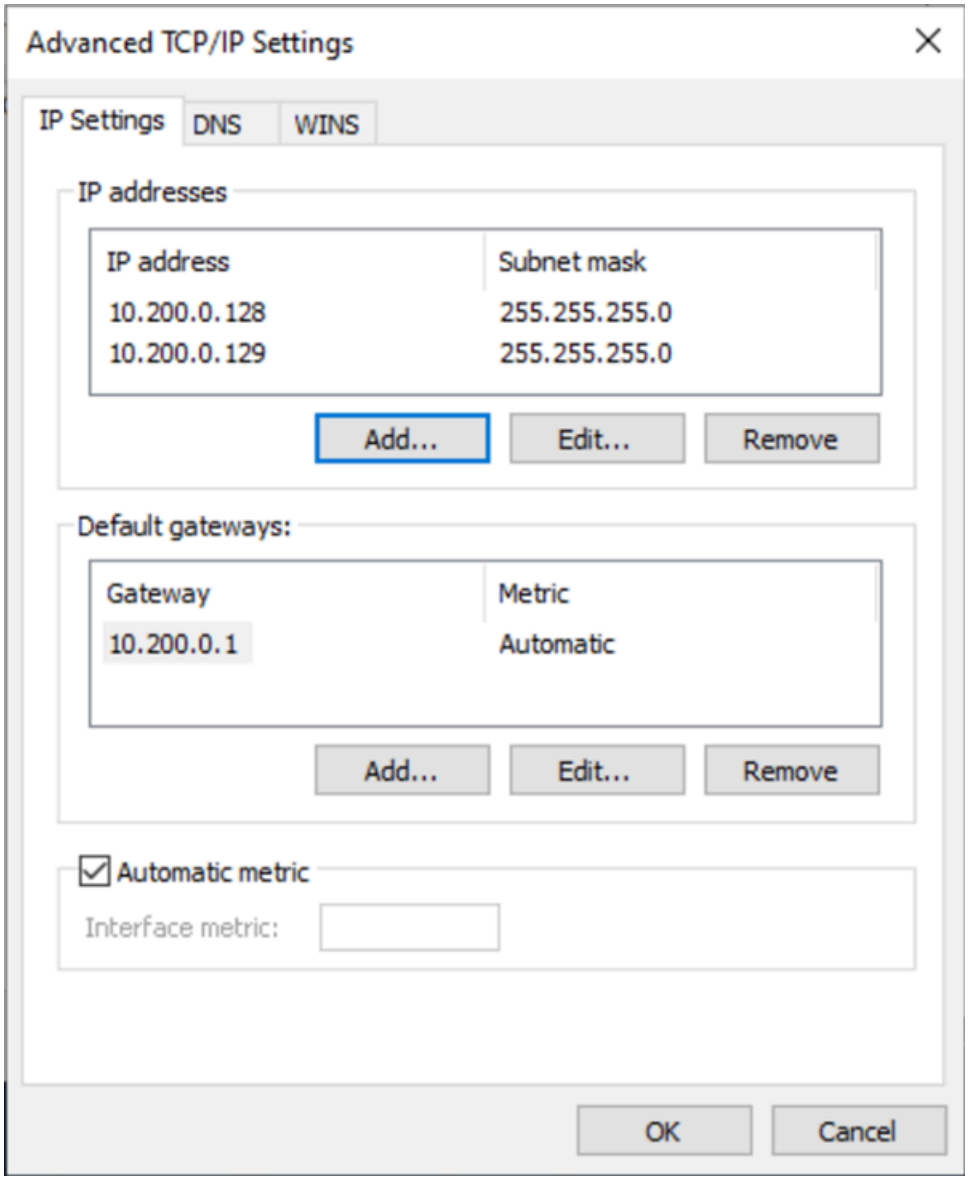
Setelah menyiapkan pengalamatan IP statis pada instans Windows, Anda siap untuk menyiapkan alamat IP privat kedua.

Untuk mengonfigurasi alamat IP sekunder

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans (dan pilih instans Anda.
3. Di bagian Jaringan, catat alamat IP sekunder.
4. Hubungkan ke instans Anda.
5. Pada instans Windows Anda, pilih Mulai, Panel Kontrol.
6. Pilih Jaringan dan Internet, Jaringan dan Pusat Berbagi.
7. Pilih antarmuka jaringan (Local Area Connection atau Ethernet) dan pilih Properties.
8. Pada halaman Pro perti Koneksi Area Lokal, pilih Internet Protocol Version 4 (TCP/IPv4), Properties, Advanced.
9. Pilih Tambahkan.
10. Di kotak dialog TCP/IP Alamat, ketik alamat IP pribadi sekunder untuk alamat IP. Untuk Subnet mask, ketik subnet mask yang sama dengan yang Anda masukkan untuk alamat IP privat primerLangkah 1: Konfigurasikan pengalamatan IP statis di instans Anda, lalu pilih Tambahkan.




11. Verifikasi pengaturan alamat IP dan pilih OK.




12. Pilih OK, Tutup.
13. Untuk mengonfirmasi bahwa alamat IP sekunder telah ditambahkan ke sistem operasi, jalankan `ipconfig /all` perintah di PowerShell. Output Anda harus menyerupai yang berikut:

```
Ethernet adapter Ethernet 4:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Amazon Elastic Network Adapter #2
   Physical Address. . . . . . . . . : 02-9C-3B-FC-8E-67
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::f4d1:a773:5afa:cd1%7(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.200.0.128(Preferred)
```

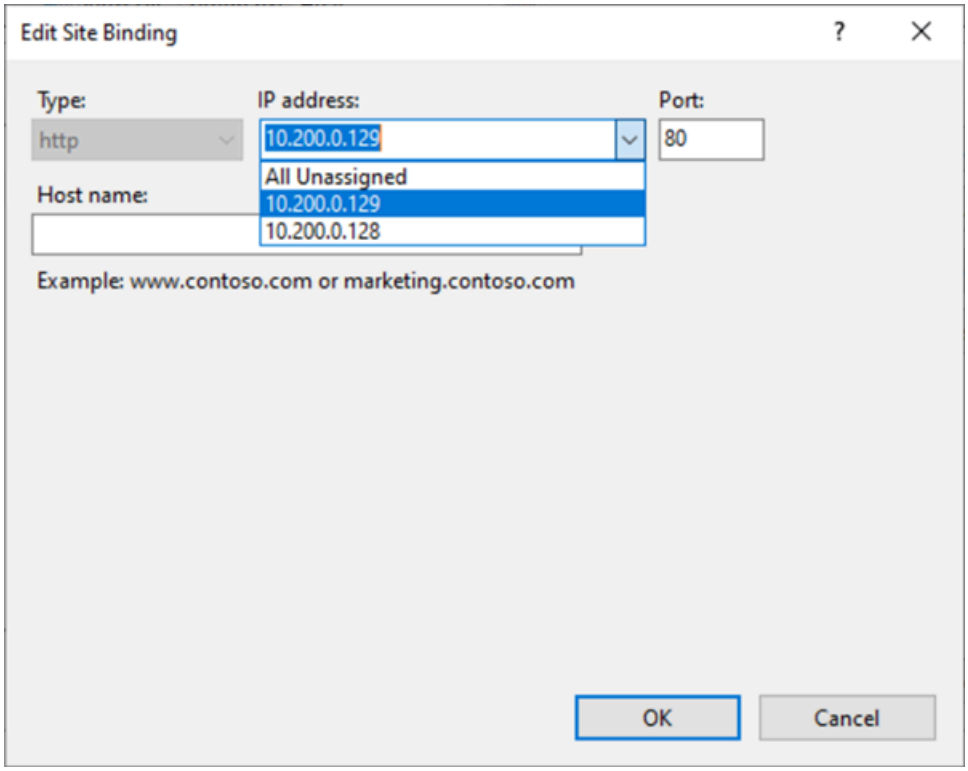
```
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   IPv4 Address. . . . . . . . . . . : 10.200.0.129(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 10.200.0.1
   DHCPv6 IAID . . . . . . . . . . . : 151166011
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2D-67-AC-FC-12-34-9A-BE-A5-E7
   DNS Servers . . . . . . . . . . . : 10.200.0.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

## Langkah 3: Konfigurasikan aplikasi untuk Menggunakan alamat IP privat sekunder

Anda dapat mengonfigurasi aplikasi apa pun untuk menggunakan alamat IP privat sekunder. Misalnya, jika instans Anda menjalankan situs web di IIS, Anda dapat mengonfigurasi IIS untuk menggunakan alamat IP privat sekunder.

Untuk mengonfigurasi IIS untuk menggunakan alamat IP privat sekunder

1. Terhubung ke instans Anda.
2. Buka Manajer Layanan Informasi Internet (IIS).
3. Di panel koneksi, perluas Situs.
4. Buka menu konteks (klik kanan) untuk situs web Anda dan pilih Edit Bindings.
5. Di kotak dialog Ikatan Situs, pada Tipe, pilih http, Edit.
6. Di kotak dialog Edit Ikatan Situs, pada Alamat IP, pilih alamat IP privat sekunder. (Secara default, setiap situs web menerima permintaan HTTP dari semua alamat IP.)




7. Pilih OK, Tutup.

# Nama host dan domain instans EC2

Memahami nama host instans EC2 dan domain penting untuk mengelola dan mengakses instans Amazon EC2 Anda secara efektif. Setiap instance EC2 dapat memiliki berbagai jenis nama host - pribadi dan publik - yang melayani tujuan yang berbeda dan mengikuti konvensi penamaan tertentu.

Topik ini menjelaskan struktur nama host instans EC2, bagaimana mereka dibangun, dan berbagai jenis nama host yang tersedia. Anda akan mempelajari cara melihat dan memodifikasi pengaturan nama host, memahami kapan harus menggunakan setiap jenis, dan mempelajari praktik terbaik untuk manajemen nama host di lingkungan Anda AWS .

Daftar Isi


- Memahami nama host instans EC2 dan domain
- Jenis nama host

# Memahami nama host instans EC2 dan domain

Alamat instance EC2 terdiri dari komponen yang berbeda. Berikut ini adalah contoh alamat instans EC2 yang menggunakan alamat IPv4 pribadi instans:

```
   IP address         Domain name
   #--------# #-----------------------#
ip-10-24-34-0.us-west-2.compute.internal
#-----------#
  Hostname
#-------------------------------------#
    Fully qualified domain name (FQDN)
```

Di mana:

- Alamat IP: Alamat IPv4 utama dari antarmuka jaringan utama yang terkait dengan sebuah instance.
- Nama host: Nama lokal dari instance EC2 tertentu (digunakan oleh sistem operasi dan untuk identifikasi jaringan lokal)
- Nama domain: Bagian dari FQDN yang menyediakan AWS
- Nama domain yang memenuhi syarat (FQDN): Alamat lengkap yang mencakup nama host dan nama domain. Ini adalah pengidentifikasi unik global lengkap yang digunakan untuk menjangkau instance Anda di seluruh jaringan.

Bergantung pada jenis nama host yang Anda pilih untuk instance atau antarmuka jaringan utama yang dilampirkan ke instance, nama host dan format nama domain akan berbeda dari contoh di atas. Bagian ini menjelaskan opsi jenis nama host.

## Jenis nama host

AWS menyediakan dua jenis nama host:  pribadi  dan  publik. Tabel berikut membandingkan perbedaan utama antara nama host pribadi dan publik, termasuk bagaimana mereka menyelesaikannya, bagaimana mereka dikonfigurasi, dan kapan harus menggunakan setiap jenis.

| | Nama host pribadi | Nama host publik |
|---|---|---|
| Resolusi DNS | Nama host pribadi mengaktifkan FQDN pribadi yang tidak dapat | Nama host publik memungkinkan FQDN publik yang dapat diakses |

| | Nama host pribadi | Nama host publik |
|---|---|---|
| | diakses dari internet publik. Nama host pribadi hanya mengizinkan permintaan untuk diselesaikan ke alamat GUA IPv4 dan IPv6 pribadi dalam VPC. | dari internet publik. Nama host publik memungkinkan permintaan untuk menyelesaikan ke GUA IPv4 dan IPv6 pribadi dalam VPC dan IP publik dari internet (DNS split-horizon). |
| Konfigurasi | Nama host pribadi dikonfigurasi pada tingkat instance. | Nama host publik dikonfigurasi pada tingkat antarmuka jaringan. |
| Kapan harus digunakan | • Untuk nama host yang diselesai kan hanya dalam VPC Anda, untuk komunikasi internal<br>• Untuk menjaga skema penamaan internal Anda tetap pribadi<br>• Untuk menyederhanakan manajemen sumber daya internal tanpa pertimbangan DNS eksternal | • Untuk mengakses instance melalui IPv4 dan IPv6 dari internet<br>• Untuk mengaktifkan migrasi yang lancar dari lingkungan IPv4 ke IPv6 |

Daftar Isi


- Nama host publik
- Nama host pribadi


## Nama host publik

Anda dapat menggunakan nama  host publik  untuk mengakses instans EC2 menggunakan nama host yang diselesaikan ke alamat IPv4 atau IPv6 publik instans dan memudahkan transisi Anda ke IPv6.

Nama host publik memungkinkan Anda untuk:

- Akses instans EC2 Anda melalui IPv4 dan IPv6, memberi Anda lebih banyak fleksibilitas dalam cara Anda terhubung ke sumber daya Anda.
- Bermigrasi dari lingkungan IPv4 ke IPv6 dengan kecepatan Anda sendiri. Anda dapat, misalnya, memisahkan migrasi database dan aplikasi, mengurangi kompleksitas dan risiko.

- Gunakan beberapa opsi nama host (IPv4-only, IPv6-only, dan dual-stack) yang secara otomatis menyelesaikan ke alamat IP yang sesuai.
- Manfaatkan peningkatan keamanan dengan DNS split-horizon, yang menyelesaikan nama host ke alamat IP pribadi saat ditanyakan dari dalam VPC Anda.

Daftar Isi


- Jenis nama host publik dan kapan menggunakannya
- Lihat nama host publik
- Ubah jenis nama host publik


## Jenis nama host publik dan kapan menggunakannya

Untuk menggunakan nama host publik, Anda harus memodifikasi antarmuka jaringan yang ada. Bagian ini menjelaskan tiga opsi jenis nama host publik dan membantu Anda memutuskan mana yang akan digunakan:

- Dual-stack - Nama berbasis IP
  - Ini adalah pilihan terbaik jika Anda bermigrasi atau berencana untuk bermigrasi dari IPv4 ke IPv6. Opsi ini memungkinkan konektivitas melalui IPv4 dan IPv6, memberikan fleksibilitas kepada klien yang mungkin menggunakan salah satu protokol dan memungkinkan klien untuk mempertahankan nama host yang sama selama migrasi ke IPv6.
  - Permintaan dari dalam VPC diselesaikan ke alamat IPv4 pribadi dan IPv6 Global Unicast Address (GUA) dari antarmuka jaringan. Permintaan dari internet diselesaikan ke IPv4 publik dan alamat GUA IPv6 dari antarmuka jaringan.
  - Contoh
    - Ketika Anda memilih opsi ini, FQDN dual-stack akan dihasilkan untuk antarmuka jaringan ini. Ini adalah contoh FQDN yang akan dihasilkan:
      - f5lnz-0khrm-nt2u3-gyqqt-nbdl5-q3cdp O.ap-southeast-2.ip.aws
    - Di mana:
      - F5lnz-0kHRM-nt2U3-GYQQT-NBDL5-Q3cdpo adalah nama host yang merupakan representasi base36 dari alamat IPv6 publik utama (f5lnz-0khrm-nt2u3-gyqqt-nbdl5) pada antarmuka jaringan bersama dengan representasi base36 dari alamat IPv4 publik primer (Q3cdpo) pada antarmuka jaringan.

    - f5lnz-0khrm-nt2u3-gyqqt-nbdl5 diselesaikan oleh Amazon DNS resolver ke alamat GUA IPv6. FFFF:1407:4:f000:81d:2689:1066:4489 Ini adalah GUA IPv6 pertama yang ditugaskan ke antarmuka jaringan.
    - Q3cdpo diselesaikan ke alamat IPv4 52.54.55.56. Ini adalah alamat IPv4 publik yang dilampirkan ke antarmuka jaringan utama.
    - ap-southeast-2 adalah Wilayah subnet tempat antarmuka jaringan berada.
    - ip.aws adalah domain yang disediakan oleh. AWS
- IPv6 - nama berbasis IP
  - Ini adalah pilihan yang baik jika Anda telah bermigrasi ke IPv6 dan memerlukan konektivitas hanya melalui IPv6.
  - Permintaan dari dalam VPC atau dari internet diselesaikan ke IPv6 GUA dari antarmuka jaringan.
  - Contoh
    - Ketika Anda memilih opsi ini, FQDN akan dihasilkan untuk antarmuka jaringan ini. Ini adalah contoh FQDN yang akan dihasilkan:
      - f5lnz-0khrm-nt2u3-gyqqt-nbdl5.ap-selatan-2.ip.aws
    - Di mana:
      - f5lnz-0khrm-nt2u3-gyqqt-nbdl5 adalah nama host yang merupakan representasi base36 dari alamat IPv6 publik utama pada antarmuka jaringan.
      - f5lnz-0khrm-nt2u3-gyqqt-nbdl5 diselesaikan oleh Amazon DNS resolver ke alamat GUA IPv6. FFFF:1407:4:f000:81d:2689:1066:4489 Ini adalah GUA IPv6 pertama yang ditugaskan ke antarmuka jaringan.
      - ap-southeast-2 adalah Wilayah subnet tempat antarmuka jaringan berada.
      - ip.aws adalah domain yang disediakan oleh. AWS
- IPv4 - nama berbasis IP
  - Ini adalah pilihan yang baik jika instance yang menggunakan antarmuka jaringan ini perlu mempertahankan akses IPv4 selama transisi ke IPv6 atau jika aplikasi atau sistem yang berjalan pada instance hanya mendukung IPv4. Ini adalah pilihan terbaik jika Anda hanya perlu mempertahankan konektivitas IPv4 dan beban kerja Anda tidak memerlukan dukungan IPv6. Misalnya, jika Anda bermigrasi ke IPv6, Anda dapat memutuskan untuk menyimpan beberapa aplikasi di IPv4 sementara yang lain pindah ke IPv6.
  - Permintaan dari dalam VPC diselesaikan ke alamat IPv4 primer pribadi dari antarmuka jaringan. Permintaan dari internet diselesaikan ke alamat IPv4 publik dari antarmuka jaringan.

- Contoh
  - Jika Anda memilih opsi ini, nama IPv4-enabled host publik akan dibuat untuk antarmuka jaringan ini. Ini adalah contoh nama DNS yang akan dihasilkan:
    - ec2-52-54-55-66.ap-southeast-2.compute.amazonaws.com
  - Di mana:
    - ec2-52-54-55-66 adalah nama host yang berasal dari alamat IPv4 publik utama dari antarmuka jaringan, menggunakan format ec2- diikuti oleh alamat IP dengan titik diganti dengan tanda hubung.
    - ec2-52-54-55-66 menyelesaikan ke alamat IPv4 52.54.55.66. Ini adalah alamat IPv4 publik yang dilampirkan ke antarmuka jaringan utama.
    - ap-southeast-2 adalah Wilayah subnet tempat antarmuka jaringan berada.
    - compute.amazonaws.com adalah domain yang disediakan oleh AWS untuk nama host publik. IPv4-based

> **Important**
>
> Dalam contoh di atas, Anda dapat melihat bahwa alamat IP digunakan untuk menghasilkan nama host. Jika Anda mengubah alamat IPv4 pribadi utama atau GUA IPv6 pertama yang ditetapkan ke antarmuka jaringan, bagian dari nama host yang diterjemahkan ke alamat IP akan berubah dan nama host publik yang dibuat sebelumnya tidak akan valid lagi. Selain itu, mengubah alamat publik IPv4 utama memaksa penyegaran hilir Instance Metadata Service (IMDS) di Panduan Pengguna Amazon EC2, sehingga metadata IMDS instans EC2 diperbarui secara otomatis.

Lihat nama host publik

Jika VPC tempat antarmuka jaringan tidak memiliki keduanya EnableDnsHostnames dan EnableDnsSupport diaktifkan, tidak ada jenis nama host yang ditentukan atau dihasilkan.

Console

Anda dapat melihat nama host publik untuk instance atau antarmuka jaringan utama.

Untuk melihat jenis nama host dan nama DNS dari sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih kotak centang untuk instans.
4. Pada tab Jaringan, di bawah Nama Host dan DNS, temukan yang berikut ini:
   - Jenis nama host publik
   - DNS Publik
   - IPv4-only Nama berbasis IP
   - IPv6-only - Nama berbasis IP
   - Dualstack - nama berbasis IP

Untuk melihat jenis nama host dan nama DNS dari antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Di bidang pencarian, masukkan ID instance. Pilih ID antarmuka jaringan untuk membuka halaman detailnya.
4. Di bawah nama host dan DNS, temukan yang berikut ini:
   - Jenis nama host publik
   - Nama DNS publik
   - Nama DNS IPv4 publik
   - Nama DNS IPv6 publik
   - Nama DNS Dualstack Publik

AWS CLI

Untuk melihat jenis nama host dan nama DNS dari antarmuka jaringan

Gunakan perintah describe-network-interfaces.

```
aws ec2 describe-network-interfaces \
    --network-interface-id eni-1234567890abcdef0 \
    --query NetworkInterfaces[].PublicIpDnsNameOptions
```

Berikut ini adalah output contoh. Karena tipe hostname adalah`public-dual-stack-dns-name`, nama host DNS sama dengan. `PublicDualStackDnsName`

```
[
     {
          "DnsHostnameType": "public-dual-stack-dns-name",
          "PublicIpv4DnsName": "ec2-52-54-55-66.ap-
southeast-2.compute.amazonaws.com",
          "PublicIpv6DnsName": "f5lnz-0khrm-nt2u3-gyqqt-nbdl5.ap-
southeast-2.ip.aws",
          "PublicDualStackDnsName": "f5lnz-0khrm-nt2u3-gyqqt-nbdl5-q3cdp0.ap-
southeast-2.ip.aws"
     }
]
```

PowerShell

Untuk melihat jenis nama host dan nama DNS dari antarmuka jaringan

Gunakan Get-EC2NetworkInterface cmdlet.

```
(Get-EC2NetworkInterface `
    -NetworkInterfaceId eni-1234567890abcdef0).PublicIpDnsNameOptions
```

Berikut ini adalah output contoh. Karena tipe hostname adalah`public-dual-stack-dns-name`, nama host DNS sama dengan. `PublicDualStackDnsName`

```
DnsHostnameType        : public-dual-stack-dns-name
PublicDualStackDnsName : f5lnz-0khrm-nt2u3-gyqqt-nbdl5-q3cdp0.ap-southeast-2.ip.aws
PublicIpv4DnsName      : ec2-52-54-55-66.ap-southeast-2.compute.amazonaws.com
PublicIpv6DnsName      : f5lnz-0khrm-nt2u3-gyqqt-nbdl5.ap-southeast-2.ip.aws
```

Ubah jenis nama host publik

Opsi jenis nama host publik bergantung pada alamat IP yang terkait dengan antarmuka jaringan:

- Jika antarmuka jaringan hanya memiliki alamat IPv4 publik, jenis nama host harus IPv4 - nama berbasis IP.
- Jika antarmuka jaringan hanya memiliki alamat IPv6, jenis nama host harus IPv6 - nama berbasis IP.
- Jika antarmuka jaringan memiliki alamat IPv4 publik dan alamat IPv6, jenis nama host dapat berupa Dual-stack - nama berbasis IP.

Prasyarat

- Antarmuka jaringan harus memiliki alamat IPv4 publik terkait atau alamat IPv6.
- VPC tempat antarmuka jaringan berada harus memiliki EnableDnsHostnames dan EnableDnsSupport diaktifkan. Lihat Melihat dan memperbarui atribut DNS untuk VPC Anda di Panduan Pengguna Amazon VPC.

Console

Untuk mengubah jenis nama host publik

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Di bidang pencarian, masukkan ID instance. Pilih kotak centang untuk antarmuka jaringan.

   Atau, dari halaman detail instance, pilih tab Jaringan dan pilih ID antarmuka jaringan untuk indeks perangkat 0.
4. Pilih T ind akan, Ubah jenis nama host publik.
5. Pilih satu opsi:

   - Dual-stack - Nama berbasis IP: Nama host publik dual-stack untuk antarmuka jaringan. Permintaan dari dalam VPC diselesaikan ke alamat IPv4 pribadi dan Alamat Unicast Global IPv6 dari antarmuka jaringan. Permintaan dari internet diselesaikan ke IPv4 publik dan alamat GUA IPv6 dari antarmuka jaringan.
   - IPv4 - Nama berbasis IP: Nama IPv4-enabled host publik untuk antarmuka jaringan. Permintaan dari dalam VPC diselesaikan ke alamat IPv4 primer pribadi dari antarmuka jaringan. Permintaan dari internet diselesaikan ke alamat IPv4 publik dari antarmuka jaringan.
   - IPv6 - Nama berbasis IP: Nama IPv6-enabled host publik untuk antarmuka jaringan. Permintaan dari dalam VPC atau dari internet diselesaikan ke IPv6 GUA dari antarmuka jaringan.
6. Pilih Ubah.

AWS CLI

Untuk mengubah jenis nama host publik

Gunakan perintah modify- public-ip-dns-name-options.

```
aws ec2 modify-public-ip-dns-name-options \
    --network-interface-id eni-1234567890abcdef0 \
    --hostname-type public-dual-stack-dns-name
```

Berikut ini adalah output contoh.

```
{
    "Successful": true
}
```

PowerShell

Untuk mengubah jenis nama host publik

Gunakan Edit-EC2PublicIpDnsNameOption cmdlet.

```
Edit-EC2PublicIpDnsNameOption `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -HostNameType public-dual-stack-dns-name
```

## Nama host pribadi

Bagian ini menjelaskan nama host pribadi instans Amazon EC2 yang tersedia saat Anda meluncurkan instans ke subnet VPC Anda.

Nama host pribadi membedakan instans EC2 di jaringan Anda. Anda dapat menggunakan nama host pribadi suatu instance jika, misalnya, Anda ingin menjalankan skrip untuk berkomunikasi dengan beberapa atau semua instance di jaringan Anda.

Daftar Isi


- Jenis nama host pribadi
- Di mana menemukan nama sumber daya dan nama IP
- Memilih antara nama sumber daya dan nama IP
- Ubah opsi penamaan berbasis sumber daya untuk Amazon EC2

Jenis nama host pribadi

Ada dua jenis nama host pribadi untuk nama host OS tamu saat instans EC2 diluncurkan di VPC:

- Nama IP: Skema penamaan warisan di mana, ketika Anda meluncurkan sebuah instans, alamat IPv4 privat dari instans tersebut disertakan dalam nama host instans. Nama IP ada selama masa pakai instans EC2. Ketika digunakan sebagai nama host DNS Privat, nama itu hanya akan mengembalikan alamat IPv4 privat (Catatan A).
- Nama sumber daya: Saat Anda meluncurkan instans, ID instans EC2 disertakan dalam nama host instans. Nama sumber daya ada selama masa pakai instans EC2. Ketika digunakan sebagai nama host DNS Pribadi, itu dapat mengembalikan alamat IPv4 pribadi (catatan A) and/or dan IPv6 Global Unicast Address (catatan AAAA).

Tipe nama host OS tamu instans EC2 bergantung pada pengaturan subnet:

- Jika instance diluncurkan ke IPv4-only subnet, Anda dapat memilih nama IP atau nama sumber daya.
- Jika instans diluncurkan ke subnet dual-stack (IPv4+IPv6), Anda dapat memilih nama IP atau nama sumber daya.
- Jika instance diluncurkan ke IPv6-only subnet, nama sumber daya digunakan secara otomatis.

Daftar Isi


- Nama IP
- Nama sumber daya
- Perbedaan antara nama IP dan nama Sumber Daya


Nama IP

Saat Anda meluncurkan instans EC2 dengan TIpe nama host dari nama IP, nama host OS tamu dikonfigurasi untuk menggunakan alamat IPv4 privat.

- Format untuk sebuah instans di us-east-1: *`private-ipv4-address`*`.ec2.internal`
- Contoh: *`ip-10-24-34-0`*`.ec2.internal`
- Format untuk instance di Wil AWS ayah lain: *`private-ipv4-address.region`*`.compute.internal`

- Contoh: *ip-10-24-34-0.us-west-2*.compute.internal

Nama sumber daya

Saat Anda meluncurkan instans EC2 di IPv6-only subnet, jenis nama host nama sumber daya dipilih secara default. Saat Anda meluncurkan instance di subnet IPv4-only atau dual-stack (IPv4+IPv6), nama sumber daya adalah opsi yang dapat Anda pilih. Setelah Anda meluncurkan sebuah instans, Anda dapat mengelola konfigurasi nama host. Untuk informasi selengkapnya, lihat Ubah opsi penamaan berbasis sumber daya untuk Amazon EC2.

Saat Anda meluncurkan instans EC2 dengan tipe nama host dari Nama sumber daya, nama host OS tamu dikonfigurasi untuk menggunakan ID instans EC2.

- Format untuk sebuah instans di us-east-1: *ec2-instance-id*.ec2.internal
- Contoh: *i-0123456789abcdef*.ec2.internal
- Format untuk instance di Wil AWS ayah lain: *ec2-instance-id.region*.compute.internal
- Contoh: *i-0123456789abcdef.us-west-2*.compute.internal

Perbedaan antara nama IP dan nama Sumber Daya

Kueri DNS untuk nama IP dan nama sumber daya hidup berdampingan untuk memastikan kompatibilitas mundur dan memungkinkan Anda bermigrasi dari penamaan berbasis IP untuk nama host ke penamaan berbasis sumber daya. Untuk nama host DNS privat berdasarkan nama IP, Anda tidak dapat mengonfigurasi apakah kueri catatan DNS A untuk instans ditanggapi atau tidak. Catatan DNS A selalu ditanggapi terlepas dari pengaturan nama host OS tamu. Sebaliknya, untuk nama host DNS pribadi berdasarkan nama sumber daya, Anda dapat mengonfigurasi apakah kueri and/or DNS A DNS AAAA untuk instance ditanggapi atau tidak. Anda mengonfigurasi perilaku respons saat meluncurkan instans atau memodifikasi subnet. Untuk informasi selengkapnya, lihat Ubah opsi penamaan berbasis sumber daya untuk Amazon EC2.

Di mana menemukan nama sumber daya dan nama IP

Anda dapat melihat jenis nama host, nama sumber daya, dan nama IP, di konsol Amazon EC2.

Daftar Isi


- Saat membuat instans EC2
- Saat melihat detail instans EC2 yang sudah ada

Saat membuat instans EC2

Saat Anda membuat instans EC2, tergantung pada tipe subnet yang Anda pilih, Tipe nama host dari Nama sumber daya mungkin tersedia atau mungkin dipilih dan tidak dapat dimodifikasi. Bagian ini menjelaskan skenario di mana Anda melihat nama sumber daya tipe nama host dan nama IP.

Skenario 1

Anda membuat instance EC2 di wizard (lihatLuncurkan instans EC2 menggunakan wizard instans peluncuran di konsol) dan, ketika Anda mengonfigurasi detail, Anda memilih subnet yang Anda konfigurasikan. IPv6-only

Dalam hal ini, Tipe nama host dari Nama sumber daya dipilih secara otomatis dan tidak dapat dimodifikasi. Opsi Nama host DNS dari Aktifkan permintaan DNS IPv4 (Catatan A) nama IP dan Aktifkan permintaan DNS IPv4 (Catatan A) berbasis sumber daya dibatalkan secara otomatis dan tidak dapat dimodifikasi. Aktifkan permintaan DNS IPv6 (catatan AAAA) berbasis sumber daya dipilih secara default, tetapi dapat dimodifikasi. Jika dipilih, permintaan DNS ke nama sumber daya akan diselesaikan ke alamat IPv6 (catatan AAAA) dari instans EC2 ini.

Skenario 2

Anda membuat instans EC2 di wizard (lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol) dan, ketika Anda mengonfigurasi detailnya, Anda memilih subnet yang dikonfigurasi dengan blok CIDR IPv4 atau blok CIDR IPv4 dan IPv6 (“tumpukan ganda”).

Dalam hal ini, Aktifkan permintaan DNS IPv4 (Catatan A) nama IP dipilih secara otomatis dan tidak dapat diubah. Ini berarti bahwa permintaan ke nama IP akan diselesaikan ke alamat IPv4 (catatan A) dari instans EC2 ini.

Opsi default ke konfigurasi subnet, tetapi Anda dapat memodifikasi opsi untuk instans ini tergantung pada pengaturan subnet:

- Tipe nama host: Menentukan apakah Anda ingin nama host OS tamu dari instans EC2 menjadi nama sumber daya atau nama IP. Nilai default-nya adalah nama IP.
- Aktifkan permintaan DNS IPv4 (catatan A) berbasis sumber daya: Menentukan apakah permintaan ke nama sumber daya Anda diselesaikan ke alamat IPv4 privat (catatan A) dari instans EC2 ini. Opsi ini tidak dipilih secara default.
- Aktifkan permintaan DNS IPv6 (catatan AAAA) berbasis sumber daya: Menentukan apakah permintaan ke nama sumber daya Anda diselesaikan ke alamat IPv6 GUA (catatan AAAA) dari instans EC2 ini. Opsi ini tidak dipilih secara default.

Saat melihat detail instans EC2 yang sudah ada

Anda dapat melihat nilai nama host untuk instans EC2 yang ada di tab Detail untuk instans EC2:

- Tipe nama host: Nama host dalam nama IP atau format nama sumber daya.
- Nama DNS IP Privat (hanya IPv4): Nama IP yang akan selalu diselesaikan ke alamat IPv4 privat dari instans.
- Nama DNS sumber daya privat: Nama sumber daya yang menyelesaikan catatan DNS yang dipilih untuk instans ini.
- Menjawab nama DNS sumber daya privat: Nama sumber daya diselesaikan ke catatan DNS IPv4 (A), IPv6 (AAAA), atau IPv4 dan IPv6 (A dan AAAA).

Selain itu, jika Anda terhubung ke instans EC2 Anda langsung melalui SSH dan memasukkan perintah `hostname`, Anda akan melihat nama host baik dalam nama IP atau format nama sumber daya.

Memilih antara nama sumber daya dan nama IP

Saat Anda meluncurkan instans EC2 (lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol), jika Anda memilih Tipe nama host dari Nama sumber daya, instans EC2 akan diluncurkan dengan nama host dalam format nama sumber daya. Dalam kasus seperti itu, catatan DNS untuk instans EC2 ini juga dapat menunjuk ke nama sumber daya. Hal ini memberi Anda fleksibilitas untuk memilih apakah nama host tersebut diselesaikan ke alamat IPv4, alamat IPv6, atau alamat IPv4 dan IPv6 dari instans tersebut. Jika Anda berencana untuk menggunakan IPv6 di masa depan atau jika Anda menggunakan subnet dual-stack hari ini, yang terbaik adalah menggunakan Tipe nama host dari Nama sumber daya sehingga Anda mengubah resolusi DNS untuk nama host instans tanpa membuat perubahan apa pun pada catatan DNS itu sendiri. Nama sumber daya memungkinkan Anda untuk menambah dan menghapus resolusi DNS IPv4 dan IPv6 pada instans EC2.

Jika sebaliknya, Anda memilih Tipe nama host dari Nama IP, dan menggunakannya sebagai nama host DNS, nama itu hanya dapat menyelesaikan ke alamat IPv4 instans. Nama ini tidak akan menyelesaikan ke alamat IPv6 dari instans bahkan jika instans memiliki alamat IPv4 dan alamat IPv6 yang terkait dengannya.

Ubah opsi penamaan berbasis sumber daya untuk Amazon EC2

Anda dapat mengubah jenis nama host dan konfigurasi nama host DNS untuk subnet, yang memengaruhi semua peluncuran instans berikutnya dalam subjek tersebut, atau Anda dapat mengubahnya untuk instans EC2 setelah Anda meluncurkannya.

Opsi penamaan berbasis sumber daya

- Jenis nama host: Menentukan pengaturan default untuk nama host OS tamu instans EC2 yang diluncurkan di subnet. Ini adalah nama sumber daya atau nama IP.
- Aktifkan permintaan IPv4 (rekaman A) nama host DNS: Menentukan apakah DNS requests/queries ke nama sumber daya diselesaikan ke alamat IPv4 pribadi (catatan A) instans EC2.
- Aktifkan permintaan DNS hostname IPv6 (rekaman AAAA): Menentukan apakah DNS requests/queries ke nama sumber daya diselesaikan ke alamat IPv6 (catatan AAAA) instans EC2.

Subnet

Mengubah pengaturan subnet tidak mengubah konfigurasi instans EC2 yang sudah diluncurkan di subnet.

Console

Untuk memodifikasi opsi untuk subnet

Buka konsol Amazon VPC dan pilih subnet. Pilih T ind akan, Edit pengaturan subnet. Ubah pengaturan sesuai kebutuhan dan kemudian simpan perubahan Anda.

AWS CLI

Untuk memodifikasi opsi untuk subnet

Gunakan perintah modify- subnet-atribut.

```
aws ec2 modify-subnet-attribute \
    --subnet-id subnet-0abcdef1234567890 \
    --private-dns-hostname-type-on-launch resource-name \
    --enable-resource-name-dns-a-record-on-launch \
    --enable-resource-name-dns-aaaa-record-on-launch
```

PowerShell

Untuk memodifikasi opsi untuk subnet

Gunakan Edit-EC2SubnetAttribute cmdlet.

```
Edit-EC2SubnetAttribute `
    -SubnetId subnet-0abcdef1234567890 `
    -PrivateDnsHostnameTypeOnLaunch ResourceName `
    -EnableResourceNameDnsAAAARecordOnLaunch $true `
    -EnableResourceNameDnsARecordOnLaunch $true
```

Instans EC2

Pertimbangan-pertimbangan

- Untuk mengubah jenis nama host, Anda harus menghentikan instance terlebih dahulu. Tidak perlu menghentikan instance untuk mengubah dua opsi lainnya.
- Karena Anda tidak dapat menghentikan instance dengan volume root penyimpanan instans, Anda hanya dapat mengonfigurasi jenis nama host dan opsi nama host DNS saat peluncuran instance. Hanya jenis instans berikut yang mendukung volume root penyimpanan instans: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

Console

Untuk mengubah jenis nama host dan opsi nama host DNS untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Jika Anda akan mengubah pengaturan Gunakan penamaan berbasis sumber daya sebagai nama host OS tamu, pertama-tama hentikan instans EC2. Jika tidak, lewati langkah ini.

   Untuk menghentikan instans, pilih instans dan pilih Status instans, Setop instans.
3. Pilih instans dan pilih Tindakan, Pengaturan instans, Ubah opsi penamaan berbasis sumber daya.

   - Gunakan penamaan berbasis sumber daya sebagai nama host OS tamu: Menentukan apakah Anda ingin nama host OS tamu dari instans EC2 menjadi nama sumber daya atau nama IP.
   - Jawab permintaan IPv4 (catatan A) nama host DNS: Menentukan apakah DNS requests/queries ke nama sumber daya Anda diselesaikan ke alamat IPv4 pribadi instans EC2 ini.

- Jawab permintaan DNS hostname IPv6 (rekaman AAAA): Menentukan apakah DNS requests/queries ke nama sumber daya Anda diselesaikan ke alamat IPv6 (catatan AAAA) instans EC2 ini.

4. Pilih Simpan.
5. Jika Anda menghentikan instans, mulai lagi.

AWS CLI

Untuk mengubah jenis nama host dan opsi nama host DNS untuk sebuah instance

Gunakan perintah modify- private-dns-name-options.

```
aws ec2 modify-private-dns-name-options \
    --instance-id i-1234567890abcdef0 \
    --private-dns-hostname-type resource-name \
    --enable-resource-name-dns-a-record \
    --enable-resource-name-dns-aaaa-record
```

PowerShell

Untuk mengubah jenis nama host dan opsi nama host DNS untuk sebuah instance

Gunakan Edit-EC2PrivateDnsNameOption cmdlet.

```
Edit-EC2PrivateDnsNameOption `
    -InstanceId i-1234567890abcdef0 `
    -PrivateDnsHostnameType ResourceName `
    -EnableResourceNameDnsAAAARecord $true`
    -EnableResourceNameDnsARecord $true
```

# Bawa alamat IP Anda sendiri (BYOIP) ke Amazon EC2

Anda dapat membawa sebagian atau semua rentang alamat IPv4 atau IPv6 yang dapat dirutekan secara publik dari jaringan lokal Anda ke jaringan Anda. Akun AWS Anda terus mengontrol rentang alamat dan Anda dapat mengiklankan rentang alamat di internet melalui AWS. Setelah Anda membawa rentang alamat ke Amazon EC2, itu muncul di Akun AWS sebagai kumpulan alamat Anda.

> **Note**
>
> Dokumentasi ini menjelaskan cara membawa rentang alamat IP Anda sendiri untuk digunakan di Amazon EC2 saja. Untuk membawa rentang alamat IP Anda sendiri untuk digunakan AWS Global Accelerator, lihat Mem bawa alamat IP Anda sendiri (BYOIP) di Panduan Peng AWS Global Accelerator embang. Untuk membawa rentang alamat IP Anda sendiri untuk digunakan Amazon VPC IP Address Manager, lihat Tutorial: Membawa alamat IP Anda ke IPAM di Panduan Pengguna Amazon VPC IPAM.

Saat Anda membawa rentang alamat IP ke AWS, AWS memvalidasi bahwa Anda mengontrol rentang alamat IP. Ada dua metode yang dapat Anda gunakan untuk menunjukkan bahwa Anda mengontrol rentang:

- Jika rentang alamat IP Anda terdaftar dengan Registry Internet yang mendukung RDAP (seperti ARIN, RIPE, dan APNIC), Anda dapat memverifikasi kontrol domain Anda dengan X.509 sertifikat dengan menggunakan proses di halaman ini. Sertifikat hanya boleh valid selama durasi proses penyediaan. Anda dapat menghapus sertifikat dari catatan RIR Anda setelah penyediaan selesai.
- Terlepas dari apakah Registry Internet Anda mendukung RDAP, Anda dapat menggunakan Amazon VPC IPAM untuk memverifikasi kontrol domain Anda dengan catatan DNS TXT. Proses itu didokumentasikan dalam Tutorial: Bawa alamat IP Anda ke IPAM di Panduan Pengguna Amazon VPC IPAM.

Untuk informasi lebih lanjut, lihat pembicar AWS aan Teknologi Online Deep Dive on Bring Your Own IP.

Daftar Isi


- Definisi BYOIP
- Persyaratan dan kuota
- Ketersediaan wilayah
- Ketersediaan Local Zone
- Prasyarat untuk BYOIP di Amazon EC2
- Masukkan rentang alamat Anda untuk digunakan di Amazon EC2
- Gunakan rentang alamat BYOIP Anda di Amazon EC2

## Definisi BYOIP

- X.509 Self-sign sertifikat — Standar sertifikat yang paling umum digunakan untuk mengenkripsi dan mengotentikasi data dalam jaringan. Ini adalah sertifikat yang digunakan oleh AWS untuk memvalidasi kontrol atas ruang IP dari catatan RDAP. Untuk informasi selengkapnya tentang X.509 sertifikat, lihat RFC 32 80.
- Nomor Sistem Otonom (ASN) — Pengidentifikasi unik global yang menentukan sekelompok prefiks IP yang dijalankan oleh satu atau lebih operator jaringan yang mempertahankan satu kebijakan perutean yang ditentukan dengan jelas.
- Regional Internet Registry (RIR) — Organisasi yang mengelola alokasi dan pendaftaran alamat IP serta ASN di wilayah dunia.
- Registry Data Access Protocol (RDAP) — Protokol read-only untuk menanyakan data registrasi saat ini dalam RIR. Entri dalam basis data RIR yang ditanyakan disebut sebagai “catatan RDAP”. Jenis catatan tertentu perlu diperbarui oleh pelanggan melalui RIR-provided mekanisme. Catatan ini ditanyakan oleh AWS untuk memverifikasi kontrol ruang alamat di RIR.
- Route Origin Authorization (ROA) — Objek yang dibuat oleh RIR bagi pelanggan untuk mengautentikasi iklan IP dalam sistem otonom tertentu. Untuk ikhtisar, lihat Route Origin Authorization (ROA) di situs web ARIN.
- Local Internet Registry (LIR) — Organisasi seperti penyedia layanan internet yang mengalokasikan blok alamat IP dari RIR untuk pelanggan mereka.

## Persyaratan dan kuota

- Rentang alamat harus terdaftar di Regional Internet Registry (RIR) Anda. Lihat RIR Anda untuk kebijakan apa pun mengenai wilayah geografis. BYOIP saat ini mendukung pendaftaran di American Registry for Internet Numbers (ARIN), Réseaux IP Européens Network Coordination Centre (RIPE), atau Asia-Pacific Pusat Informasi Jaringan (APNIC). Rentang alamat ini harus didaftarkan untuk entitas bisnis atau kelembagaan dan tidak dapat didaftarkan untuk perorangan.
- Rentang alamat IPv4 paling spesifik yang dapat Anda bawa adalah /24.
- Rentang alamat IPv6 paling spesifik yang dapat Anda bawa adalah /48 untuk CIDR yang dapat diiklankan secara publik dan /60 untuk CIDR yang tidak dapat diiklankan secara publik. ???
- ROA tidak diperlukan untuk rentang CIDR yang tidak dapat diiklankan secara publik, tetapi catatan RDAP masih perlu diperbarui.
- Anda dapat membawa setiap rentang alamat ke satu AWS Wilayah pada satu waktu.

- Anda dapat membawa total lima rentang alamat BYOIP IPv4 dan IPv6 per Wil AWS ayah ke akun Anda. AWS Anda tidak dapat menyesuaikan kuota untuk CIDR BYOIP menggunakan konsol Kuota Layanan, tetapi Anda dapat meminta peningkatan kuota dengan menghubungi Pusat AWS Dukungan seperti yang dijelaskan dalam kuota AWS layanan di. Referensi Umum AWS
- Anda tidak dapat membagikan rentang alamat IP Anda dengan akun lain menggunakan AWS RAM kecuali Anda menggunakan Amazon VPC IP Address Manager (IPAM) dan mengintegrasikan IPAM dengan Organisasi. AWS Untuk informasi selengkapnya, lihat Mengintegrasikan IPAM dengan AWS Organisasi di Panduan Pengguna Amazon VPC IPAM.
- Alamat dalam rentang alamat IP harus memiliki riwayat yang bersih. Kami mungkin menginvestigasi reputasi rentang alamat IP dan berhak menolak rentang alamat IP jika berisi alamat IP yang memiliki reputasi buruk atau terkait dengan perilaku jahat.
- Ruang alamat warisan, ruang alamat IPv4 yang didistribusikan oleh registri pusat Internet Assigned Numbers Authority (IANA) sebelum pembentukan sistem Regional Internet Registry (RIR), masih membutuhkan objek ROA yang sesuai.
- Untuk LIR, biasanya mereka menggunakan proses manual untuk memperbarui catatan mereka. Deployment bisa memakan waktu berhari-hari, tergantung pada LIR.
- Objek ROA tunggal dan catatan RDAP diperlukan untuk blok CIDR yang besar. Anda dapat membawa beberapa blok CIDR yang lebih kecil dari rentang itu ke AWS, bahkan di beberapa AWS Wilayah, menggunakan objek dan catatan tunggal.
- BYOIP tidak didukung untuk Zona Panjang Gelombang atau aktif. AWS Outposts
- Jangan membuat perubahan manual untuk BYOIP di RADB atau IRR lainnya. BYOIP akan secara otomatis memperbarui RADB. Setiap perubahan manual yang menyertakan ASN BYOIP akan menyebabkan operasi penyediaan BYOIP gagal.
- Setelah Anda membawa rentang alamat IPv4 ke AWS, Anda dapat menggunakan semua alamat IP dalam rentang, termasuk alamat pertama (alamat jaringan) dan alamat terakhir (alamat siaran).

## Ketersediaan wilayah

Fitur BYOIP saat ini tersedia di semua Wilayah AWS komersial kecuali untuk Wilayah Tiongkok.

## Ketersediaan Local Zone

Zona Lokal adalah perpanjangan dari Wil AWS ayah dalam kedekatan geografis dengan pengguna Anda. Local Zones dikelompokkan ke dalam “grup perbatasan jaringan”. Di dalam AWS, grup perbatasan jaringan adalah kumpulan Zona Ketersediaan (AZ), Zona Lokal, atau Zona Panjang

Gelombang yang mengik AWS lankan alamat IP publik. Zona Lokal mungkin memiliki grup perbatasan jaringan yang berbeda dari AZ di Wil AWS ayah untuk memastikan latensi minimum atau jarak fisik antara AWS jaringan dan pelanggan yang mengakses sumber daya di Zona ini.

Anda dapat menyediakan rentang alamat BYOIPv4 dan mengiklankannya di grup perbatasan jaringan Local Zone berikut menggunakan opsi `--network-border-group`:

- af-selatan-1-los-1
- ap-timur laut-1-tpe-1
- ap-selatan-1-ccu-1
- ap-selatan-1-del-1
- ap-South-1-bkk-1
- ap-South-1-mnl-1
- ap-South-2-akl-1
- ap-Southeast-2-per-1
- eu-central-1-ham-1
- eu-central-1-ist-1
- eu-central-1-waw-1
- eu-utara-1-cph-1
- eu-utara-1-hel-1
- saya-selatan-1-mct-1
- us-timur-1-atl-2
- us-timur-1-bos-1
- us-timur-1-bue-1
- us-timur-1-chi-2
- us-east-1-dfw-2
- us-timur-1-iah-2
- us-timur-1-lim-1
- us-timur-1-mci-1
- us-timur-1-mia-2
- us-timur-1-msp-1

- us-timur-1-nyc-1
- us-timur-1-nyc-2
- us-timur-1-phl-1
- us-timur-1-qro-1
- us-timur-1-scl-1
- us-barat-2-den-1
- us-barat-2-hnl-1
- us-barat-2-las-1
- us-west-2-lax-1
- us-barat-2-pdx-1
- us-west-2-phx-2
- us-barat-2-laut-1

Jika Local Zones diaktifkan (lihat Mengaktifkan Local Zone), Anda dapat memilih grup perbatasan jaringan untuk Local Zones saat menyediakan dan mengiklankan CIDR BYOIPv4. Pilih grup perbatasan jaringan dengan hati-hati karena EIP dan AWS sumber daya yang terkait dengannya harus berada di grup perbatasan jaringan yang sama.

> **Note**
>
> Anda tidak dapat menyediakan atau mengiklankan rentang alamat BYOIPv6 di Zona Lokal saat ini, kecuali untuk`eu-central-1-ist-1`, yang mendukung BYOIPv4 dan BYOIPv6.

Di Zona `eu-central-1-ist-1` Lokal, iklan dan penarikan adalah operasi asinkron yang dapat memakan waktu hingga lima hari kerja untuk menyelesaikannya. Saat Anda mengiklankan rentang alamat, statusnya `provisioned` berubah dari ke`pending-advertising`, lalu ke `advertised` setelah operasi selesai. Saat Anda menariknya, status berubah dari `advertised` ke`pending-withdrawal`, lalu kembali ke`provisioned`. Anda dapat memantau kemajuan operasi ini melalui Kon AWS sol atau CLI.

Jika Anda menggunakan ASN Anda sendiri (BYOASN), Anda dapat menentukannya saat mengiklankan rentang alamat Anda. Anda tidak dapat mengubah ASN yang terkait dengan CIDR setelah iklan. Untuk mengubah ASN, tarik CIDR dan beriklan kembali.

> **Important**
>
> Anda tidak dapat membatalkan iklan atau penarikan yang tertunda. Anda harus menunggu operasi selesai sebelum memulai operasi berikutnya.

## Prasyarat untuk BYOIP di Amazon EC2

Proses orientasi untuk BYOIP memiliki dua fase, di mana Anda harus melakukan tiga langkah. Langkah-langkah ini sesuai dengan langkah-langkah yang digambarkan dalam diagram berikut. Kami menyertakan langkah-langkah manual dalam dokumentasi ini, tetapi RIR Anda mungkin menawarkan layanan terkelola untuk membantu Anda dengan langkah-langkah ini.

> **Tip**
>
> Tugas-tugas pada bagian ini memerlukan terminal Linux dan dapat dijalankan menggunakan Linux, AWS CloudShell, atau Windows Subsystem for Linux.

Daftar Isi


- Gambaran umum
- Buat kunci pribadi dan buat X.509 sertifikat
- Unggah X.509 sertifikat ke catatan RDAP di RIR Anda
- Membuat objek ROA di RIR Anda


### Gambaran umum

Fase persiapan

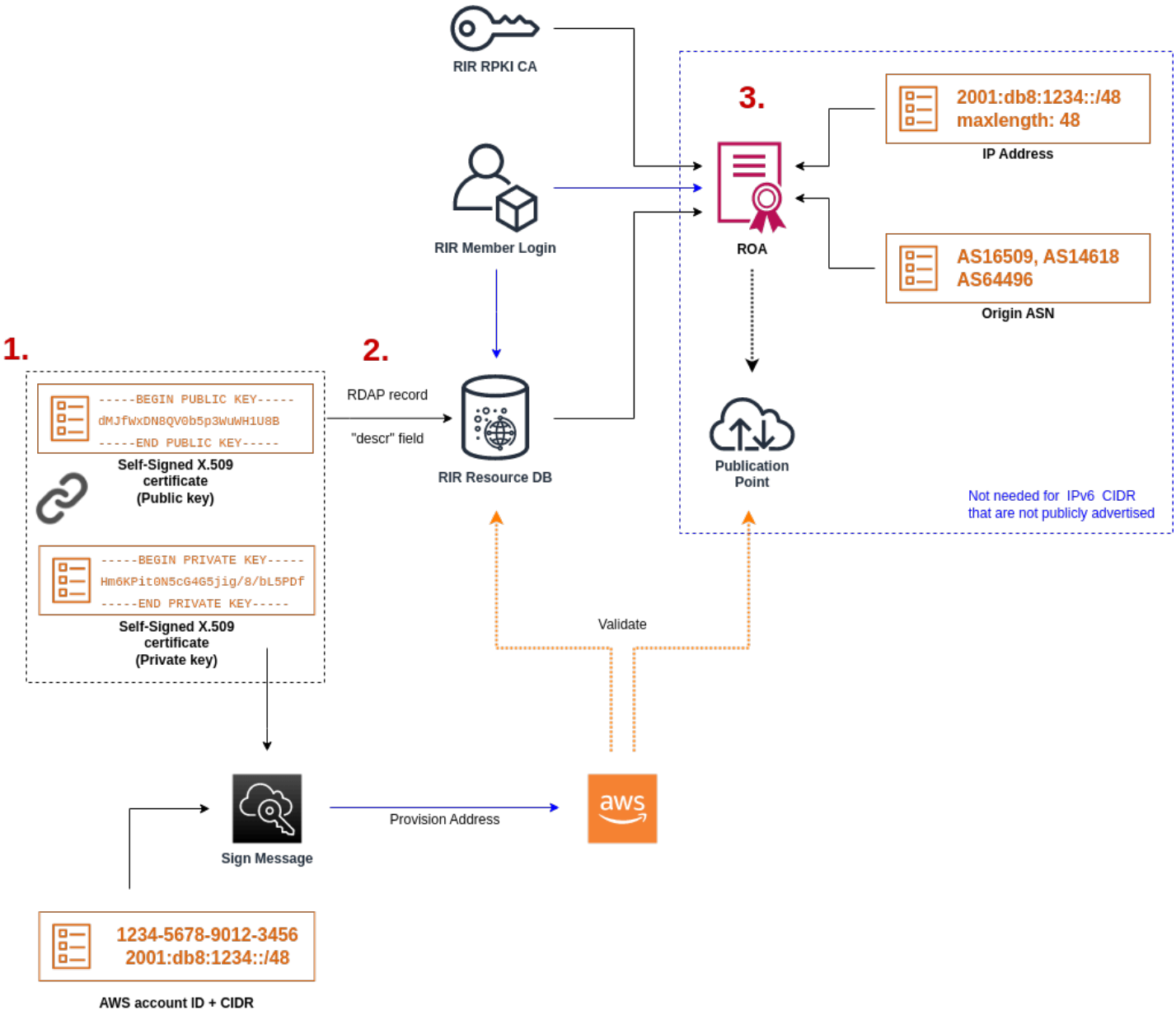
[1] Buat kunci pribadi dan gunakan untuk menghasilkan X.509 sertifikat yang ditandatangani sendiri untuk tujuan otentikasi. Sertifikat ini hanya digunakan selama fase penyediaan. Anda dapat menghapus sertifikat dari catatan RIR Anda setelah penyediaan selesai

Fase konfigurasi RIR

[2] Unggah sertifikat yang ditandatangani sendiri ke komentar catatan RDAP Anda.

[3] Buat objek ROA di RIR Anda. ROA menentukan rentang alamat yang diinginkan, Nomor Sistem Otonom (ASN) diizinkan untuk mengiklankan rentang alamat, dan tanggal kedaluwarsa untuk mendaftar ke Resource Public Key Infrastructure (RPKI) RIR Anda.

> **Note**
>
> ROA tidak diperlukan untuk ruang alamat IPv6 yang tidak dapat diiklankan secara publik.



Untuk membawa beberapa rentang alamat yang tidak berdekatan, Anda harus mengulangi proses ini dengan setiap rentang alamat. Namun, langkah persiapan dan konfigurasi RIR tidak perlu diulang jika memisahkan blok yang berdekatan di beberapa Wilayah yang berbeda. AWS

Membawa rentang alamat tidak berpengaruh pada setiap rentang alamat yang Anda bawa sebelumnya.

### Buat kunci pribadi dan buat X.509 sertifikat

Gunakan prosedur berikut untuk membuat X.509 sertifikat yang ditandatangani sendiri dan menambahkannya ke catatan RDAP untuk RIR Anda. Pasangan kunci ini digunakan untuk mengautentikasi rentang alamat dengan RIR. Perintah openssl memerlukan OpenSSL versi 1.0.2 atau lebih baru.

Salin perintah berikut dan mengganti nilai placeholder saja (dalam teks miring berwarna).

Prosedur ini mengikuti praktik terbaik mengenkripsi kunci RSA privat Anda dan memerlukan frasa sandi untuk mengaksesnya.

1. Buat kunci privat RSA 2048 bit seperti yang ditunjukkan berikut ini.

   ```
   $ openssl genpkey -aes256 -algorithm RSA -pkeyopt rsa_keygen_bits:2048 -out
    private-key.pem
   ```

   Parameter `-aes256` menentukan algoritma yang digunakan untuk mengenkripsi kunci privat. Perintah mengembalikan output berikut, termasuk petunjuk untuk mengatur frasa sandi:

   ```
   ......+++
   .+++
   Enter PEM pass phrase: xxxxxxx
   Verifying - Enter PEM pass phrase: xxxxxxx
   ```

   Anda dapat memeriksa kunci menggunakan perintah berikut:

   ```
   $ openssl pkey -in private-key.pem -text
   ```

   Ini mengembalikan prompt frasa-sandi dan isi kunci, yang harus mirip dengan berikut ini:

   ```
   Enter pass phrase for private-key.pem: xxxxxxx
   -----BEGIN PRIVATE KEY-----
   ```

```
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQDFBXHRI4HVKAhh
3seiciooizCRTbJe1+YsxNTja4XyKypVGIFWDGhZs44FCHlPOOSVJ+NqP74w96oM
7DPS3xo9kaQyZBFn2YEp2EBq5vf307KHNRmZZUmkn0zHOSEpNmY2fMxISBxewlxR
FAniwmSd/8TDvHJMY9FvAIvWuTsv5l0tJKk+a91K4+tO3UdDR7Sno5WXExfsBrW3
g1ydo3TBsx8i5/YiVOcNApy7ge2/FiwY3aCXJB6r6nuF6H8mRgI4r4vkMRsOlAhJ
DnZPNeweboo+K3Q3lwbgbmOKD/z9svk8N/+hUTBtIX0fRtbG+PLIw3xWRHGrMSn2
BzsPVuDLAgMBAAECggEACiJUj2hfJkKv47Dc3es3Zex67A5uDVjXmxfox2Xhdupn
fAcNqAptV6fXt0SPUNbhUxbBKNbshoJGufFwXPli1SXnpzvkdU4Hyco4zgbhXFsE
RNYjYfOGzTPwdBLpNMB6k3Tp4RHse6dNrlH0jDhpioL8cQEBdBJyVF5X0wymEbmV
mC0jgH/MxsBAPWW6ZKicg9ULMlWiAZ3MRAZPjHHgpYkAAsUWKAbCBwVQcVjGO59W
jfZjzTX5pQtVVH68ruciH88DTZCwjCkjBhxg+OIkJBLE5wkh82jIHSivZ63flwLw
z+E0+HhELSZJrn2MY6Jxmik3qNNUOF/Z+3msdj2luQKBgQDjwlC/3jxp8zJy6P8o
JQKv7TdvMwUj4VSWOHZBHLv4evJaaia0uQjIo1UDa8AYitqhX1NmCCehGH8yuXj/
v6V3CzMKDkmRr1NrONnSz5QsndQ04Z6ihAQlPmJ96g4wKtgoC7AYpyP0g1a+4/sj
b1+o3YQI4pD/F71c+qaztH7PRwKBgQDdc23yNmT3+Jyptf0fKjEvONK+xwUKzi9c
L/OzBq5yOIC1Pz2T85gOe1i8kwZws+xlpG6uBT6lmIJELd0k59FyupNu4dPvX5SD
6GGqdx4jk9KvI74usGeOBohmF0phTHkrWKBxXiyT0oS8zjnJlEn8ysIpGgO28jjr
LpaHNZ/MXQKBgQDfLNcnS0LzpsS2aK0tzyZU8SMyqVHOGMxj7quhneBq2T6FbiLD
T9TVlYaGNZ0j71vQaLI19qOubWymbautH0Op5KV8owdf4+bf1/NJaPIOzhDUSIjD
Qo01WW31Z9XDSRhKFTnWzmCjBdeIcajyzf10YKsycaAW9lItu8aBrMndnQKBgQDb
nNp/JyRwqjOrNljk7DHEs+SD39kHQzzCfqd+dnTPv2sc06+cpym3yulQcbokULpy
fmRo3bin/pvJQ3aZX/Bdh9woTXqhXDdrrSwWInVYMQPyPk8f/D9mIOJp5FUWMwHD
U+whIZSxsEeE+jtixlWtheKRYkQmzQZXbWdIhYyI3QKBgD+F/6wcZ85QW8nAUykA
3WrSIx/3cwDGdm4NRGct8ZOZjTHjiy9ojMOD1L7iMhRQ/3k3hUsin5LDMp/ryWGG
x4uIaLat40kiC7T4I66DM7P59euqdz3w0PD+VU+h7GSivvsFDdySUt7bNK0AUVLh
dMJfWxDN8QV0b5p3WuWH1U8B
-----END PRIVATE KEY-----
Private-Key: (2048 bit)
modulus:
    00:c5:05:71:d1:23:81:d5:28:08:61:de:c7:a2:72:
    2a:28:8b:30:91:4d:b2:5e:d7:e6:2c:c4:d4:e3:6b:
    85:f2:2b:2a:55:18:81:56:0c:68:59:b3:8e:05:08:
    79:4f:38:e4:95:27:e3:6a:3f:be:30:f7:aa:0c:ec:
    33:d2:df:1a:3d:91:a4:32:64:11:67:d9:81:29:d8:
    40:6a:e6:f7:f7:d3:b2:87:35:19:99:65:49:a4:9f:
    4c:c7:39:21:29:36:66:36:7c:cc:48:48:1c:5e:c2:
    5c:51:14:09:e2:c2:64:9d:ff:c4:c3:bc:72:4c:63:
    d1:6f:00:8b:d6:b9:3b:2f:e6:5d:2d:24:a9:3e:6b:
    dd:4a:e3:eb:4e:dd:47:43:47:b4:a7:a3:95:97:13:
    17:ec:06:b5:b7:83:5c:9d:a3:74:c1:b3:1f:22:e7:
    f6:22:54:e7:0d:02:9c:bb:81:ed:bf:16:2c:18:dd:
    a0:97:24:1e:ab:ea:7b:85:e8:7f:26:46:02:38:af:
    8b:e4:31:1b:0e:94:08:49:0e:76:4f:35:ec:1e:6e:
    8a:3e:2b:74:37:97:06:e0:6e:63:8a:0f:fc:fd:b2:
```

```
    f9:3c:37:ff:a1:51:30:6d:21:7d:1f:46:d6:c6:f8:
    f2:c8:c3:7c:56:44:71:ab:31:29:f6:07:3b:0f:56:
    e0:cb
publicExponent: 65537 (0x10001)
privateExponent:
    0a:22:54:8f:68:5f:26:42:af:e3:b0:dc:dd:eb:37:
    65:ec:7a:ec:0e:6e:0d:58:d7:9b:17:e8:c7:65:e1:
    76:ea:67:7c:07:0d:a8:0a:6d:57:a7:d7:b7:44:8f:
    50:d6:e1:53:16:c1:28:d6:ec:86:82:46:b9:f1:70:
    5c:f9:62:d5:25:e7:a7:3b:e4:75:4e:07:c9:ca:38:
    ce:06:e1:5c:5b:04:44:d6:23:61:f3:86:cd:33:f0:
    74:12:e9:34:c0:7a:93:74:e9:e1:11:ec:7b:a7:4d:
    ae:51:f4:8c:38:69:8a:82:fc:71:01:01:74:12:72:
    54:5e:57:d3:0c:a6:11:b9:95:98:2d:23:80:7f:cc:
    c6:c0:40:3d:65:ba:64:a8:9c:83:d5:0b:32:55:a2:
    01:9d:cc:44:06:4f:8c:71:e0:a5:89:00:02:c5:16:
    28:06:c2:07:05:50:71:58:c6:3b:9f:56:8d:f6:63:
    cd:35:f9:a5:0b:55:54:7e:bc:ae:e7:22:1f:cf:03:
    4d:90:b0:8c:29:23:06:1c:60:f8:e2:24:24:12:c4:
    e7:09:21:f3:68:c8:1d:28:af:67:ad:df:97:02:f0:
    cf:e1:34:f8:78:44:2d:26:49:ae:7d:8c:63:a2:71:
    9a:29:37:a8:d3:54:38:5f:d9:fb:79:ac:76:3d:a5:
    b9
prime1:
    00:e3:c2:50:bf:de:3c:69:f3:32:72:e8:ff:28:25:
    02:af:ed:37:6f:33:05:23:e1:54:96:38:76:41:1c:
    bb:f8:7a:f2:5a:6a:26:b4:b9:08:c8:a3:55:03:6b:
    c0:18:8a:da:a1:5f:53:66:08:27:a1:18:7f:32:b9:
    78:ff:bf:a5:77:0b:33:0a:0e:49:91:af:53:6b:38:
    d9:d2:cf:94:2c:9d:d4:34:e1:9e:a2:84:04:25:3e:
    62:7d:ea:0e:30:2a:d8:28:0b:b0:18:a7:23:f4:83:
    56:be:e3:fb:23:6f:5f:a8:dd:84:08:e2:90:ff:17:
    bd:5c:fa:a6:b3:b4:7e:cf:47
prime2:
    00:dd:73:6d:f2:36:64:f7:f8:9c:a9:b5:fd:1f:2a:
    31:2f:38:d2:be:c7:05:0a:ce:2f:5c:2f:f3:b3:06:
    ae:72:38:80:b5:3f:3d:93:f3:98:0e:7b:58:bc:93:
    06:70:b3:ec:65:a4:6e:ae:05:3e:a5:98:82:44:2d:
    dd:24:e7:d1:72:ba:93:6e:e1:d3:ef:5f:94:83:e8:
    61:aa:77:1e:23:93:d2:af:23:be:2e:b0:67:8e:06:
    88:66:17:4a:61:4c:79:2b:58:a0:71:5e:2c:93:d2:
    84:bc:ce:39:c9:94:49:fc:ca:c2:29:1a:03:b6:f2:
    38:eb:2e:96:87:35:9f:cc:5d
exponent1:
```

```
    00:df:2c:d7:27:4b:42:f3:a6:c4:b6:68:ad:2d:cf:
    26:54:f1:23:32:a9:51:ce:18:cc:63:ee:ab:a1:9d:
    e0:6a:d9:3e:85:6e:22:c3:4f:d4:d5:95:86:86:35:
    9d:23:ef:5b:d0:68:b2:35:f6:a3:ae:6d:6c:a6:6d:
    ab:ad:1f:43:a9:e4:a5:7c:a3:07:5f:e3:e6:df:d7:
    f3:49:68:f2:0e:ce:10:d4:48:88:c3:42:8d:35:59:
    6d:f5:67:d5:c3:49:18:4a:15:39:d6:ce:60:a3:05:
    d7:88:71:a8:f2:cd:fd:74:60:ab:32:71:a0:16:f6:
    52:2d:bb:c6:81:ac:c9:dd:9d
exponent2:
    00:db:9c:da:7f:27:24:70:aa:33:ab:36:58:e4:ec:
    31:c4:b3:e4:83:df:d9:07:43:3c:c2:7e:a7:7e:76:
    74:cf:bf:6b:1c:d3:af:9c:a7:29:b7:ca:e9:50:71:
    ba:24:50:ba:72:7e:64:68:dd:b8:a7:fe:9b:c9:43:
    76:99:5f:f0:5d:87:dc:28:4d:7a:a1:5c:37:6b:ad:
    2c:16:22:75:58:31:03:f2:3e:4f:1f:fc:3f:66:20:
    e2:69:e4:55:16:33:01:c3:53:ec:21:21:94:b1:b0:
    47:84:fa:3b:62:c6:55:ad:85:e2:91:62:44:26:cd:
    06:57:6d:67:48:85:8c:88:dd
coefficient:
    3f:85:ff:ac:1c:67:ce:50:5b:c9:c0:53:29:00:dd:
    6a:d2:23:1f:f7:73:00:c6:76:6e:0d:44:67:2d:f1:
    93:99:8d:31:e3:8b:2f:68:8c:c3:83:d4:be:e2:32:
    14:50:ff:79:37:85:4b:22:9f:92:c3:32:9f:eb:c9:
    61:86:c7:8b:88:68:b6:ad:e3:49:22:0b:b4:f8:23:
    ae:83:33:b3:f9:f5:eb:aa:77:3d:f0:d0:f0:fe:55:
    4f:a1:ec:64:a2:be:fb:05:0d:dc:92:52:de:db:34:
    ad:00:51:52:e1:74:c2:5f:5b:10:cd:f1:05:74:6f:
    9a:77:5a:e5:87:d5:4f:01
```

Simpan kunci privat Anda di lokasi yang aman saat tidak digunakan.

2. Hasilkan X.509 sertifikat menggunakan kunci pribadi yang dibuat pada langkah sebelumnya. Dalam contoh ini, sertifikat kedaluwarsa dalam 365 hari, setelahnya sertifikat tidak dapat dipercaya. Pastikan Anda mengatur waktu kedaluwarsa dengan tepat. Sertifikat hanya boleh valid selama durasi proses penyediaan. Anda dapat menghapus sertifikat dari catatan RIR Anda setelah penyediaan selesai. Perintah `tr -d "\n"` menghapus karakter baris baru (jeda baris) dari output. Anda harus memberikan Nama Umum saat diminta, tetapi bidang lainnya dapat dibiarkan kosong.

```
$ openssl req -new -x509 -key private-key.pem -days 365 | tr -d "\n" >
 certificate.pem
```

Ini menghasilkan output serupa dengan yang berikut ini:

```
Enter pass phrase for private-key.pem: xxxxxxx
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) []:
State or Province Name (full name) []:
Locality Name (eg, city) []:
Organization Name (eg, company) []:
Organizational Unit Name (eg, section) []:
Common Name (eg, fully qualified host name) []:example.com
Email Address []:
```

> **Note**
>
> Nama Umum tidak diperlukan untuk penyedi AWS aan. Itu bisa berupa nama domain internal atau publik.

Anda dapat memeriksa sertifikat dengan perintah berikut:

```
$ cat certificate.pem
```

Outputnya harus berupa PEM-encoded string panjang tanpa jeda baris, diawali oleh `-----BEGIN CERTIFICATE-----` dan diikuti oleh`-----END CERTIFICATE-----`.

## Unggah X.509 sertifikat ke catatan RDAP di RIR Anda

Tambahkan sertifikat yang telah Anda buat sebelumnya ke catatan RDAP untuk RIR Anda. Pastikan untuk memasukkan string `-----BEGIN CERTIFICATE-----` dan `-----END CERTIFICATE-----` sebelum dan sesudah bagian yang dikodekan. Semua konten ini harus dalam satu baris panjang. Prosedur untuk memperbarui RDAP tergantung pada RIR Anda:

- Untuk ARIN, gunakan portal Manajer Akun untuk menambahkan sertifikat di bagian “Komentar Publik” untuk objek “Informasi Jaringan” yang mewakili rentang alamat Anda. Jangan menambahkannya ke bagian komentar untuk organisasi Anda.
- Untuk RIPE, tambahkan sertifikat sebagai bidang “descr” baru ke objek “inetnum” atau “inet6num” yang mewakili rentang alamat Anda. Ini biasanya dapat ditemukan di bagian “Sumber Daya Saya” di portal Basis Data RIPE. Jangan menambahkannya ke bagian komentar untuk organisasi Anda atau bidang “komentar” dari objek di atas.
- Untuk APNIC, tambahkan sertifikat dengan mengedit komentar pada catatan “inetnum” atau “inet6num”.

Anda dapat menghapus sertifikat dari catatan RIR Anda setelah tahap penyediaan di bawah ini selesai.

### Membuat objek ROA di RIR Anda

Buat objek ROA untuk mengotorisasi Amazon ASN 16509 dan 14618 untuk mengiklankan rentang alamat Anda, ditambah yang saat ini berwenang untuk mengiklankan rentang alamat. Untuk AWS GovCloud (US) Regions, otorisasi hanya ASN 8987. Untuk AWS European Sovereign Cloud, otorisasi ASN 16509 dan 214101. Anda harus mengatur panjang maksimum ke ukuran CIDR yang Anda bawa. Prefiks IPv4 paling spesifik yang dapat Anda bawa adalah /24. Rentang alamat IPv6 paling spesifik yang dapat Anda bawa adalah /48 untuk CIDR yang dapat diiklankan secara publik dan /60 untuk CIDR yang tidak dapat diiklankan secara publik.

> **Important**
>
> Jika Anda membuat objek ROA untuk Manajer Alamat IP (IPAM) Amazon VPC, saat Anda membuat ROA, untuk IPv4 CIDR Anda harus mengatur panjang maksimum prefiks alamat IP ke `/24`. Untuk IPv6 CIDR, jika Anda menambahkannya ke kolam yang dapat diiklankan, panjang maksimum prefiks alamat IP harus. `/48` Ini memastikan bahwa Anda memiliki fleksibilitas penuh untuk membagi alamat IP publik Anda di seluruh Wil AWS ayah. IPAM memberlakukan panjang maksimum yang Anda tetapkan. Untuk informasi selengkapnya tentang alamat BYOIP ke IPAM, lihat Tutorial: CIDR alamat BYOIP ke IPAM di Panduan Pengguna Amazon VPC IPAM.

Ini mungkin perlu waktu hingga 24 jam agar ROA tersedia di Amazon. Untuk informasi lebih lanjut, konsultasikan RIR Anda:

- ARIN — Permintaan ROA
- RIPE — Mengelola ROAs
- APNIC — Manajemen Rute

Saat Anda memigrasikan iklan dari beban kerja lokal ke AWS, Anda harus membuat ROA untuk ASN yang ada sebelum membuat ROA untuk ASN Amazon. Jika tidak, Anda mungkin melihat dampak pada perutean dan iklan yang ada.

**Important**

Agar Amazon dapat mengiklankan dan terus mengiklankan rentang alamat IP Anda, ROA dengan ASN Amazon harus sesuai dengan pedoman di atas. Jika ROA Anda tidak valid atau tidak sesuai dengan pedoman di atas, Amazon berhak untuk berhenti mengiklankan rentang alamat IP Anda.

**Note**

Langkah ini tidak diperlukan untuk ruang alamat IPv6 yang tidak dapat diiklankan secara publik.

## Masukkan rentang alamat Anda untuk digunakan di Amazon EC2

Proses orientasi untuk BYOIP mencakup tugas-tugas berikut, tergantung pada kebutuhan Anda.

Tugas

- Menyediakan rentang alamat yang dapat diiklankan secara publik di AWS
- Menyediakan rentang alamat IPv6 yang tidak dapat diiklankan secara publik
- Iklankan rentang alamat melalui AWS
- Mencabut akses rentang alamat
- Validasi BYOIP Anda

## Menyediakan rentang alamat yang dapat diiklankan secara publik di AWS

Saat Anda menyediakan rentang alamat untuk digunakan AWS, Anda mengonfirmasi bahwa Anda mengontrol rentang alamat dan memberi wewenang kepada Amazon untuk mengiklankannya. Kami juga memverifikasi bahwa Anda mengontrol rentang alamat melalui pesan otorisasi yang ditandatangani. Pesan ini ditandatangani dengan pasangan X.509 kunci yang ditandatangani sendiri yang Anda gunakan saat memperbarui catatan RDAP dengan sertifikat. X.509 AWS membutuhkan pesan otorisasi yang ditandatangani secara kriptografis yang disajikan ke RIR. RIR mengotentikasi tanda tangan terhadap sertifikat yang ditambahkan ke RDAP, dan memeriksa rincian otorisasi terhadap ROA.

Untuk menyediakan rentang alamat

1. Membuat pesan

   Menulis pesan otorisasi teks biasa. Format pesan adalah sebagai berikut, di mana tanggal tersebut adalah tanggal kedaluwarsa pesan:

   ```
   1|aws|account|cidr|YYYYMMDD|SHA256|RSAPSS
   ```

   Ganti nomor akun, rentang alamat, dan tanggal kedaluwarsa dengan nilai Anda sendiri untuk membuat pesan yang menyerupai hal berikut ini:

   ```
   text_message="1|aws|0123456789AB|198.51.100.0/24|20211231|SHA256|RSAPSS"
   ```

   Ini agar tidak menjadi bingung dengan pesan ROA, yang memiliki tampilan serupa.

2. Menandatangani pesan

   Menandatangani pesan teks biasa menggunakan kunci privat yang telah Anda buat sebelumnya. Tanda tangan yang dikembalikan oleh perintah ini adalah string panjang yang perlu Anda gunakan pada langkah berikutnya.

   > **Important**
   >
   > Kami sarankan Anda menyalin dan menempelkan perintah ini. Kecuali untuk isi pesan, jangan mengubah atau mengganti nilai apa pun.

```
signed_message=$( echo -n $text_message | openssl dgst -sha256 -sigopt
 rsa_padding_mode:pss -sigopt rsa_pss_saltlen:-1 -sign private-key.pem -keyform PEM
 | openssl base64 | tr -- '+=/' '-_~' | tr -d "\n")
```

3. Penyediaan alamat

   Gunakan perintah AWS CLI provision-byoip-cidr untuk menyediakan rentang alamat. Opsi `--cidr-authorization-context` menggunakan string pesan dan tanda tangan yang telah Anda buat sebelumnya.

   > **Important**
   >
   > Anda harus menentukan Wil AWS ayah tempat rentang BYOIP harus disediakan jika berbeda dari Konfigurasi. AWS CLI `Default region name`

   ```
   aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context
    Message="$text_message",Signature="$signed_message" --region us-east-1
   ```

   Penyediaan rentang alamat adalah operasi asinkron, sehingga panggilan segera kembali, tetapi rentang alamat belum siap untuk digunakan hingga statusnya berubah dari `pending-provision` menjadi `provisioned`.

4. Memantau kemajuan

   Meskipun sebagian besar penyediaan akan selesai dalam waktu dua jam, mungkin diperlukan waktu hingga satu minggu untuk menyelesaikan proses penyediaan untuk rentang yang dapat diiklankan secara publik. Gunakan perintah describe-byoip-cidrs untuk memantau kemajuan, seperti dalam contoh ini:

   ```
   aws ec2 describe-byoip-cidrs --max-results 5 --region us-east-1
   ```

   Jika ada masalah selama penyediaan dan status masuk ke `failed-provision`, Anda harus menjalankan perintah `provision-byoip-cidr` lagi setelah masalah terpecahkan.

## Menyediakan rentang alamat IPv6 yang tidak dapat diiklankan secara publik

Secara default, rentang alamat disediakan agar dapat diiklankan secara publik ke internet. Anda dapat menyediakan rentang alamat IPv6 yang tidak akan dapat diiklankan secara publik. Untuk rute yang tidak dapat diakses secara publik, proses penyediaan umumnya selesai dalam hitungan menit. Saat Anda mengaitkan blok IPv6 CIDR dari rentang alamat non-publik dengan VPC, IPv6 CIDR hanya dapat diakses melalui opsi konektivitas hibrida yang mendukung IPv6, seperti, AWS Site-to-Site VPN Direct Connect, atau Amazon VPC Transit Gateway. https://docs.aws.amazon.com/vpc/latest/tgw/what-is-transit-gateway.html

ROA tidak diperlukan untuk menyediakan rentang alamat non-publik.

> **Important**
>
> - Anda hanya dapat menentukan apakah rentang alamat dapat diiklankan secara publik selama penyediaan. Anda tidak dapat mengubah status yang dapat diiklankan dari rentang alamat di lain waktu.
> - Amazon VPC tidak mendukung CIDR alamat lokal unik (ULA). Semua VPC harus memiliki IPv6 CIDR yang unik. Dua VPC tidak dapat memiliki rentang IPv6 CIDR yang sama.

Untuk menyediakan rentang alamat IPv6 yang tidak akan dapat diiklankan secara publik, gunakan perintah provision-byoip-cidr.

```
aws ec2 provision-byoip-cidr --cidr address-range --cidr-authorization-context
 Message="$text_message",Signature="$signed_message" --no-publicly-advertisable --
region us-east-1
```

## Iklankan rentang alamat melalui AWS

Setelah rentang alamat disediakan, rentang alamat tersebut siap untuk diiklankan. Anda harus mengiklankan rentang alamat persis yang Anda sediakan. Anda tidak dapat mengiklankan hanya sebagian dari rentang alamat yang disediakan.

Jika Anda menyediakan rentang alamat IPv6 yang tidak akan diiklankan secara publik, Anda tidak perlu menyelesaikan langkah ini.

Kami menyarankan Anda berhenti mengiklankan rentang alamat atau bagian mana pun dari rentang dari lokasi lain sebelum Anda mengiklankannya AWS. Jika Anda terus mengiklankan rentang

alamat IP Anda atau bagiannya dari lokasi lain, kami tidak dapat mendukungnya dengan andal atau memecahkan masalah. Secara khusus, kami tidak dapat menjamin bahwa lalu lintas ke rentang alamat atau sebagian rentang akan memasuki jaringan kami.

Untuk meminimalkan waktu henti, Anda dapat mengonfigurasi AWS sumber daya untuk menggunakan alamat dari kumpulan alamat sebelum diiklankan, dan kemudian secara bersamaan berhenti mengiklankannya dari lokasi saat ini dan mulai mengiklankannya AWS. Untuk informasi lebih lanjut tentang pengalokasian alamat IP Elastis dari kumpulan alamat Anda, lihat Mengalokasikan alamat IP Elastis.

Batasan

- Anda dapat menjalankan perintah advertise-byoip-cidr maksimal sekali setiap 10 detik, meskipun Anda menentukan rentang alamat yang berbeda setiap kali melakukannya.
- Anda dapat menjalankan perintah withdraw-byoip-cidr maksimal sekali setiap 10 detik, meskipun Anda menentukan rentang alamat yang berbeda setiap kali melakukannya.

Untuk mengiklankan rentang alamat, gunakan perintah advertise-byoip-cidr berikut.

```
aws ec2 advertise-byoip-cidr --cidr address-range --region us-east-1
```

Untuk berhenti mengiklankan rentang alamat, gunakan perintah withdraw-byoip-cidr berikut.

```
aws ec2 withdraw-byoip-cidr --cidr address-range --region us-east-1
```

> **Note**
>
> Di Zona `eu-central-1-ist-1` Lokal, iklan dan penarikan tidak sinkron. Rentang alamat memasuki `pending-advertising` `pending-withdrawal` status atau sebelum operasi selesai. Untuk informasi selengkapnya, lihat Ketersediaan Zona Lokal.

## Mencabut akses rentang alamat

Untuk berhenti menggunakan rentang alamat Anda AWS, pertama-tama lepaskan alamat IP Elastic apa pun dan pisahkan blok IPv6 CIDR yang masih dialokasikan dari kumpulan alamat. Kemudian, hentikan iklan rentang alamat, dan terakhir, cabut akses rentang alamat.

Anda tidak dapat mencabut akses sebagian rentang alamat. Jika Anda ingin menggunakan rentang alamat yang lebih spesifik dengan AWS, hapus ketentuan seluruh rentang alamat dan sediakan rentang alamat yang lebih spesifik.

(IPv4) Untuk melepas setiap alamat IP Elastis, gunakan perintah release-address berikut.

```
aws ec2 release-address --allocation-id eipalloc-12345678abcabcabc --region us-east-1
```

(IPv6) Untuk memisahkan blok CIDR IPv6, gunakan perintah berikut disassociate-vpc-cidr-block.

```
aws ec2 disassociate-vpc-cidr-block --association-id vpc-cidr-assoc-12345abcd1234abc1
 --region us-east-1
```

Untuk berhenti mengiklankan rentang alamat, gunakan perintah withdraw-byoip-cidr berikut.

```
aws ec2 withdraw-byoip-cidr --cidr address-range --region us-east-1
```

Untuk membatalkan rentang alamat, gunakan perintah deprovision-byoip-cidr berikut.

```
aws ec2 deprovision-byoip-cidr --cidr address-range --region us-east-1
```

Diperlukan waktu hingga satu hari untuk mencabut akses rentang alamat.

## Validasi BYOIP Anda

1. Validasi pasangan kunci x.509 yang ditandatangani sendiri

   Memvalidasi bahwa sertifikat telah diunggah dan valid melalui perintah whois.

   Untuk ARIN, gunakan `whois -h whois.arin.net r + 2001:0DB8:6172::/48` untuk mencari catatan RDAP untuk rentang alamat Anda. Periksa `Public Comments` bagian untuk `NetRange` (rentang jaringan) di output perintah. Sertifikat harus ditambahkan di `Public Comments` bagian untuk rentang alamat.

   Anda dapat memeriksa sertifikat yang `Public Comments` berisi menggunakan perintah berikut:

   ```
   whois -h whois.arin.net r + 2001:0DB8:6172::/48 | grep Comments | grep BEGIN
   ```

   Ini mengembalikan output dengan isi kunci, yang harus mirip dengan berikut ini:

```
Public Comments:
-----BEGIN CERTIFICATE-----
MIID1zCCAr+gAwIBAgIUBkRPNSLrPqbRAFP8RDAHSP+I1TowDQYJKoZIhvcNAQE
LBQAwezELMAkGA1UEBhMCTloxETAPBgNVBAgMCEF1Y2tsYW5kMREwDwYDVQQHDA
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFdlYiBTZXJ2aWNlczETMBEGA1UEC
wwKQllPSVAgRGVtbzETMBEGA1UEAwwKQllPSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpvbiBXZWIgU2
VydmljZXMxEzARBgNVBAsMCkJZT0lQIERlbW8xEzARBgNVBAMMCkJZT0lQIERlb
W8wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCfmacvDp0wZ0ceiXXc
R/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanAEskgAseyFypwEEQr4CJijI/5hp9
prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3stsI5QesHVRwOaXUdprAnndaTug
mDPkD0vrl475JWDSIm+PUxGWLy+60aBqiaZq35wU/x+wXlAqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp1ZnVIc7NqnhdeIW48QaYjhMlUEf
xdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2rGlHWkJsbhr0VEUyAGu1bwkgcdww
3A7NjOxQbAgMBAAGjUzBRMB0GA1UdDgQWBBStFyujN6SYBr2glHpGt0XGF7GbGT
AfBgNVHSMEGDAWgBStFyujN6SYBr2glHpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIb3DQEBCwUAA4IBAQBX6nn6YLhz521lfyVfxY0t6o3410bQAeAF
08ud+ICtmQ4IO4A4B7zV3zIVYr0clrOOaFyLxngwMYN0XY5tVhDQqk4/gmDNEKS
Zy2QkX4Eg0YUWVzOyt6fPzjOvJLcsqc1hcF9wySL507XQz76Uk5cFypBOzbnk35
UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGCvRDl/qdO/GIDJi77dmZWkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIsoNPyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```

Untuk RIPE, gunakan `whois -r -h whois.ripe.net` *`2001:0DB8:7269::/48`* untuk mencari catatan RDAP untuk rentang alamat Anda. Periksa bagian `descr` untuk objek `inetnum` (rentang jaringan) di output perintah. Sertifikat harus ditambahkan sebagai bidang `descr` baru untuk rentang alamat.

Anda dapat memeriksa sertifikat yang `descr` berisi menggunakan perintah berikut:

```
whois -r -h whois.ripe.net 2001:0DB8:7269::/48 | grep descr | grep BEGIN
```

Ini mengembalikan output dengan isi kunci, yang harus mirip dengan berikut ini:

```
descr:
-----BEGIN CERTIFICATE-----MIID1zCCAr+gAwIBAgIUBkRPNSLrPqbRAFP8
RDAHSP+I1TowDQYJKoZIhvcNAQELBQAwezELMAkGA1UEBhMCTloxETAPBgNVBAg
MCEF1Y2tsYW5kMREwDwYDVQQHDAhBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIF
```

```
dlYiBTZXJ2aWNlczETMBEGA1UECwwKQllPSVAgRGVtbzETMBEGA1UEAwwKQllPS
VAgRGVtbzAeFw0yMTEyMDcyMDI0NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNV
BAYTAk5aMREwDwYDVQQIDAhBdWNrbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDA
aBgNVBAoME0FtYXpvbiBXZWIgU2VydmljZXMxEzARBgNVBAsMCkJZT0lQIERlbW
8xEzARBgNVBAMMCkJZT0lQIERlbW8wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwg
gEKAoIBAQCfmacvDp0wZ0ceiXXcR/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanA
EskgAseyFypwEEQr4CJijI/5hp9prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3
stsI5QesHVRwOaXUdprAnndaTugmDPkD0vrl475JWDSIm+PUxGWLy+60aBqiaZq
35wU/x+wXlAqBXg4MZK2KoUu27kYt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp
1ZnVIc7NqnhdeIW48QaYjhMlUEfxdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2r
GlHWkJsbhr0VEUyAGu1bwkgcdww3A7NjOxQbAgMBAAGjUzBRMB0GA1UdDgQWBBS
tFyujN6SYBr2glHpGt0XGF7GbGTAfBgNVHSMEGDAWgBStFyujN6SYBr2glHpGt0
XGF7GbGTAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBCwUAA4IBAQBX6nn6Y
Lhz521lfyVfxY0t6o3410bQAeAF08ud+ICtmQ4IO4A4B7zV3zIVYr0clrOOaFyL
xngwMYN0XY5tVhDQqk4/gmDNEKSZy2QkX4Eg0YUWVzOyt6fPzjOvJLcsqc1hcF9
wySL507XQz76Uk5cFypBOzbnk35UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8
mBGqVpPpey+dXpzzzv1iBKN/VY4ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGC
vRDl/qdO/GIDJi77dmZWkh/ic90MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIsoN
PyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```

Untuk APNIC, gunakan `whois -h whois.apnic.net` *`2001:0DB8:6170::/48`* untuk mencari catatan RDAP untuk rentang alamat BYOIP Anda. Periksa bagian `remarks` untuk objek `inetnum` (rentang jaringan) di output perintah. Sertifikat harus ditambahkan sebagai bidang `remarks` baru untuk rentang alamat.

Anda dapat memeriksa sertifikat yang `remarks` berisi menggunakan perintah berikut:

```
whois -h whois.apnic.net 2001:0DB8:6170::/48 | grep remarks | grep BEGIN
```

Ini mengembalikan output dengan isi kunci, yang harus mirip dengan berikut ini:

```
remarks:
-----BEGIN CERTIFICATE-----
MIID1zCCAr+gAwIBAgIUBkRPNSLrPqbRAFP8RDAHSP+I1TowDQYJKoZIhvcNAQE
LBQAwezELMAkGA1UEBhMCTloxETAPBgNVBAgMCEF1Y2tsYW5kMREwDwYDVQQHDA
hBdWNrbGFuZDEcMBoGA1UECgwTQW1hem9uIFdlYiBTZXJ2aWNlczETMBEGA1UEC
wwKQllPSVAgRGVtbzETMBEGA1UEAwwKQllPSVAgRGVtbzAeFw0yMTEyMDcyMDI0
NTRaFw0yMjEyMDcyMDI0NTRaMHsxCzAJBgNVBAYTAk5aMREwDwYDVQQIDAhBdWN
rbGFuZDERMA8GA1UEBwwIQXVja2xhbmQxHDAaBgNVBAoME0FtYXpvbiBXZWIgU2
VydmljZXMxEzARBgNVBAsMCkJZT0lQIERlbW8xEzARBgNVBAMMCkJZT0lQIERlb
W8wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQCfmacvDp0wZ0ceiXXc
```

```
R/q27mHI/U5HKt7SST4X2eAqufR9wXkfNanAEskgAseyFypwEEQr4CJijI/5hp9
prh+jsWHWwkFRoBRR9FBtwcU/45XDXLga7D3stsI5QesHVRwOaXUdprAnndaTug
mDPkD0vrl475JWDSIm+PUxGWLy+60aBqiaZq35wU/x+wXlAqBXg4MZK2KoUu27k
Yt2zhmy0S7Ky+oRfRJ9QbAiSu/RwhQbh5Mkp1ZnVIc7NqnhdeIW48QaYjhMlUEf
xdaqYUinzz8KpjfADZ4Hvqj9jWZ/eXo/9b2rGlHWkJsbhr0VEUyAGu1bwkgcdww
3A7NjOxQbAgMBAAGjUzBRMB0GA1UdDgQWBBStFyujN6SYBr2glHpGt0XGF7GbGT
AfBgNVHSMEGDAWgBStFyujN6SYBr2glHpGt0XGF7GbGTAPBgNVHRMBAf8EBTADA
QH/MA0GCSqGSIb3DQEBCwUAA4IBAQBX6nn6YLhz521lfyVfxY0t6o3410bQAeAF
08ud+ICtmQ4IO4A4B7zV3zIVYr0clrOOaFyLxngwMYN0XY5tVhDQqk4/gmDNEKS
Zy2QkX4Eg0YUWVzOyt6fPzjOvJLcsqc1hcF9wySL507XQz76Uk5cFypBOzbnk35
UkWrzA9KK97cXckfIESgK/k1N4ecwxwG6VQ8mBGqVpPpey+dXpzzzv1iBKN/VY4
ydjgH/LBfdTsVarmmy2vtWBxwrqkFvpdhSGCvRDl/qdO/GIDJi77dmZWkh/ic90
MNk1f38gs1jrCj8lThoar17Uo9y/Q5qJIsoNPyQrJRzqFU9F3FBjiPJF
-----END CERTIFICATE-----
```

2. Validasi pembuatan objek ROA

   Validasi keberhasilan pembuatan objek ROA menggunakan API RIPEstat Data. Pastikan untuk menguji rentang alamat Anda terhadap Amazon ASNs 16509 dan 14618, ditambah ASNs yang saat ini berwenang untuk mengiklankan rentang alamat. Untuk AWS GovCloud (US) Regions, uji terhadap ASN 8987. Untuk AWS European Sovereign Cloud, uji terhadap ASN 16509 dan 214101.

   Anda dapat memeriksa objek ROA dari ASN Amazon yang berbeda dengan rentang alamat Anda dengan menggunakan perintah berikut:

   ```
   curl --location --request GET "https://stat.ripe.net/data/rpki-validation/data.json?
   resource=ASN&prefix=CIDR
   ```

   Dalam contoh output ini, respons memiliki hasil `"status": "valid"` untuk Amazon ASN 16509. Ini menunjukkan objek ROA untuk rentang alamat berhasil dibuat:

   ```
   {
       "messages": [],
       "see_also": [],
       "version": "0.3",
       "data_call_name": "rpki-validation",
       "data_call_status": "supported",
       "cached": false,
       "data": {
           "validating_roas": [
               {
                   "origin": "16509",
   ```

```
                "prefix": "2001:0DB8::/32",
                "max_length": 48,
                "validity": "valid"
            },
            {
                "origin": "14618",
                "prefix": "2001:0DB8::/32",
                "max_length": 48,
                "validity": "invalid_asn"
            },
            {
                "origin": "64496",
                "prefix": "2001:0DB8::/32",
                "max_length": 48,
                "validity": "invalid_asn"
            }
        ],
        "status": "valid",
        "validator": "routinator",
        "resource": "16509",
        "prefix": "2001:0DB8::/32"
    },
    "query_id": "20230224152430-81e6384e-21ba-4a86-852a-31850787105f",
    "process_time": 58,
    "server_id": "app116",
    "build_version": "live.2023.2.1.142",
    "status": "ok",
    "status_code": 200,
    "time": "2023-02-24T15:24:30.773654"
}
```


Status “unknown” menunjukkan objek ROA untuk rentang alamat belum dibuat. Status “invalid_asn” menunjukkan bahwa objek ROA untuk rentang alamat tidak berhasil dibuat.

## Gunakan rentang alamat BYOIP Anda di Amazon EC2

Anda dapat melihat serta menggunakan rentang alamat IPv4 dan IPv6 yang telah Anda sediakan di akun Anda. Untuk informasi selengkapnya, lihat the section called “Masukkan rentang alamat Anda”.

## Rentang alamat IPv4

Anda dapat membuat alamat IP Elastic dari kumpulan alamat IPv4 dan menggunakannya dengan AWS sumber daya Anda, seperti instans EC2, gateway NAT, dan Network Load Balancers.

Untuk melihat informasi tentang kumpulan alamat IPv4 berikut ini yang telah Anda sediakan di akun Anda, gunakan perintah describe-public-ipv4-pools.

```
aws ec2 describe-public-ipv4-pools --region us-east-1
```

Untuk membuat alamat IP Elastis dari kumpulan alamat IPv4, gunakan perintah allocate-address. Anda dapat menggunakan opsi `--public-ipv4-pool` untuk menentukan ID dari kumpulan alamat yang dikembalikan oleh `describe-byoip-cidrs`. Atau Anda dapat menggunakan opsi `--address` untuk menentukan alamat dari rentang alamat yang Anda sediakan.

## Rentang alamat IPv6

Untuk melihat informasi tentang kumpulan alamat IPv6 berikut ini yang telah Anda sediakan di akun Anda, gunakan perintah describe-ipv6-pools berikut.

```
aws ec2 describe-ipv6-pools --region us-east-1
```

Untuk membuat VPC dan menentukan CIDR IPv6 dari kumpulan alamat IPv6, gunakan berikut perintah create-vpc berikut. Untuk mengizinkan Amazon memilih CIDR IPv6 dari kumpulan alamat IPv6, hilangkan opsi `--ipv6-cidr-block`.

```
aws ec2 create-vpc --cidr-block 10.0.0.0/16 --ipv6-cidr-block ipv6-cidr --ipv6-
pool pool-id --region us-east-1
```

Untuk mengaitkan blok CIDR IPv6 dari kelompok alamat IPv6 Anda dengan VPC, gunakan perintah associate-vpc-cidr-block berikut. Untuk mengizinkan Amazon memilih CIDR IPv6 dari kumpulan alamat IPv6, hilangkan opsi `--ipv6-cidr-block`.

```
aws ec2 associate-vpc-cidr-block --vpc-id vpc-123456789abc123ab --ipv6-cidr-block ipv6-
cidr --ipv6-pool pool-id --region us-east-1
```

Untuk melihat VPC Anda dan informasi kumpulan alamat IPv6, gunakan perintah describe-vpcs. Untuk melihat informasi tentang blok CIDR IPv6 dari kumpulan alamat IPv6 tertentu, gunakan perintah get-associated-ipv6-pool-cidrs berikut.

```
aws ec2 get-associated-ipv6-pool-cidrs --pool-id pool-id --region us-east-1
```

Jika Anda memisahkan blok CIDR IPv6 dari VPC Anda, ini akan dilepas kembali ke kumpulan alamat IPv6 Anda.

# Alamat Elastic IP

Alamat IP Elastis adalah sebuah alamat IPv4 statis untuk komputasi cloud dinamis. Alamat IP Elastic dialokasikan ke AWS akun Anda, dan menjadi milik Anda sampai Anda melepaskannya. Dengan alamat IP Elastis, Anda dapat menutupi kegagalan suatu instans atau perangkat lunak dengan meremajakan secara cepat alamat ke instans lain di akun Anda. Atau, Anda dapat menentukan alamat IP Elastis dalam catatan DNS untuk domain Anda, sehingga domain Anda menunjuk ke instans Anda. Untuk informasi selengkapnya, lihat dokumentasi untuk registrar domain Anda.

Alamat IP Elastis adalah alamat IPv4 publik, yang dapat dijangkau dari internet. Jika Anda perlu terhubung ke instance yang tidak memiliki alamat IPv4 publik, Anda dapat mengaitkan alamat IP Elastic dengan instance Anda untuk mengaktifkan komunikasi dengan internet.

Daftar Isi


- Harga alamat IP Elastis
- Dasar alamat IP Elastis
- Kuota alamat IP Elastis
- Mengaitkan alamat IP Elastis dengan instans
- Mentransfer alamat IP elastis antara Akun AWS
- Merilis alamat IP Elastis
- Buat catatan DNS terbalik untuk email di Amazon EC2


## Harga alamat IP Elastis

Ada biaya untuk semua alamat IP Elastic apakah mereka sedang digunakan (dialokasikan ke sumber daya, seperti instans EC2) atau idle (dibuat di akun Anda tetapi tidak dialokasikan).

AWS biaya untuk semua alamat IPv4 publik, termasuk alamat IPv4 publik yang terkait dengan instans yang sedang berjalan dan alamat IP Elastic. Untuk informasi selengkapnya, lihat tab Alamat IPv4 Publik di halaman harga Amazon VPC.

# Dasar alamat IP Elastis

Berikut adalah karakteristik dasar dari alamat IP Elastis:

- Alamat IP Elastis bersifat statis; alamat ini tidak berubah seiring waktu.
- Alamat IP Elastis hanya untuk digunakan di Wilayah tertentu saja, dan tidak dapat dipindahkan ke Wilayah yang berbeda.
- Alamat IP Elastic berasal dari kumpulan alamat IPv4 Amazon, atau dari kumpulan alamat IPv4 khusus yang telah Anda bawa ke alamat Anda. Akun AWS Kami tidak mendukung alamat IP Elastis untuk IPv6.
- Untuk menggunakan alamat IP Elastis, pertama-tama Anda mengalokasikannya ke akun Anda, lalu mengaitkannya dengan instans atau antarmuka jaringan.
- Ketika Anda mengaitkan alamat IP Elastis dengan sebuah instans, alamat ini juga dikaitkan dengan antarmuka jaringan primer instans tersebut. Ketika Anda mengaitkan alamat IP Elastis dengan sebuah antarmuka jaringan yang ditambahkan ke sebuah instans, alamat ini juga dikaitkan dengan instans tersebut.
- Saat Anda mengaitkan alamat IP Elastic dengan instance atau antarmuka jaringan utamanya, jika instans sudah memiliki alamat IPv4 publik yang terkait dengannya, alamat IPv4 publik tersebut dilepaskan kembali ke kumpulan alamat IPv4 publik Amazon dan alamat IP Elastic dikaitkan dengan instance sebagai gantinya. Anda tidak dapat menggunakan kembali alamat IPv4 publik yang sebelumnya terkait dengan instance dan Anda tidak dapat mengonversi alamat IPv4 publik tersebut menjadi alamat IP Elastic. Untuk informasi selengkapnya, lihat Alamat IPv4 publik.
- Anda dapat memisahkan alamat IP Elastis dari sumber daya, kemudian mengaitkannya dengan sumber daya yang berbeda. Untuk menghindari perilaku yang tidak terduga, pastikan semua koneksi aktif ke sumber daya yang disebutkan dalam kaitan yang ada ditutup sebelum Anda melakukan perubahan. Setelah Anda mengaitkan alamat IP Elastis Anda ke sumber daya yang berbeda, Anda dapat membuka kembali koneksi Anda ke sumber daya yang baru saja dikaitkan.
- Alamat IP Elastis yang tidak terkait tetap dialokasikan ke akun Anda hingga Anda secara eksplisit melepasnya. Anda akan dikenakan biaya untuk semua alamat IP Elastic di akun Anda, terlepas dari apakah alamat tersebut terkait atau tidak terkait dengan instance. Untuk informasi selengkapnya, lihat tab Alamat IPv4 Publik di halaman harga Amazon VPC.
- Saat Anda mengaitkan alamat IP Elastis dengan sebuah instans yang sebelumnya memiliki alamat IPv4 publik, nama host DNS publik instans tersebut berubah untuk mencocokkan dengan alamat IP Elastis.

- Kami menyelesaikan nama host DNS publik untuk alamat IPv4 publik atau alamat IP Elastis dari instans di luar jaringan instans, dan untuk alamat IPv4 privat instans dari dalam jaringan instans.
- Ketika Anda mengalokasikan alamat IP Elastic dari kumpulan alamat IP yang telah Anda bawa ke AWS akun Anda, itu tidak diperhitungkan dalam batas alamat IP Elastic Anda. Untuk informasi selengkapnya, lihat Kuota alamat IP Elastis.
- Saat Anda mengalokasikan alamat IP Elastis, Anda dapat mengaitkan alamat IP Elastis dengan grup border jaringan. Ini adalah lokasi tempat kita mengiklankan blok CIDR. Mengatur grup perbatasan jaringan membatasi blok CIDR ke grup ini. Jika Anda tidak menentukan grup batas jaringan, kami menetapkan grup batas yang berisi semua Zona Ketersediaan di Wilayah tersebut (misalnya, `us-west-2`).
- Alamat IP Elastis hanya untuk digunakan dalam grup batas jaringan tertentu.

## Kuota alamat IP Elastis

Secara default, semua Akun AWS memiliki kuota lima (5) alamat IP elastis per Wilayah, karena alamat internet publik (IPv4) adalah sumber daya publik yang langka. Kami sangat menyarankan Anda menggunakan alamat IP Elastic terutama untuk kemampuannya untuk memetakan ulang alamat ke instance lain jika terjadi kegagalan instance, dan untuk menggunakan nama host DNS untuk semua komunikasi antar-node lainnya.

Jika menurut Anda arsitektur Anda menjamin alamat IP Elastis tambahan, Anda dapat meminta peningkatan kuota secara langsung dari konsol Kuota Layanan. Untuk meminta kenaikan kuota, pilih Permintaan peningkatan di tingkat akun. Untuk informasi selengkapnya, lihat Kuota layanan Amazon EC2.

## Mengaitkan alamat IP Elastis dengan instans

Setelah mengalokasikan alamat IP Elastic, Anda dapat mengaitkannya dengan AWS sumber daya, seperti instance EC2, gateway NAT, atau Network Load Balancer. Untuk mengaitkan alamat IP Elastic dengan AWS sumber daya yang berbeda nanti, Anda dapat memisahkannya dari sumber daya saat ini dan kemudian menghubungkannya dengan sumber daya baru.

Tugas

- Mengalokasikan alamat IP Elastis
- Kaitkan sebuah alamat IP Elastis
- Pisahkan alamat IP Elastis

## Mengalokasikan alamat IP Elastis

Anda dapat mengalokasikan alamat IP Elastic untuk digunakan di Wilayah. Ada biaya untuk semua alamat IP Elastic apakah mereka sedang digunakan (terkait dengan sumber daya, seperti instans EC2) atau idle (dibuat di akun Anda tetapi tidak terkait).

Console

Untuk mengalokasikan Alamat IP elastis

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Jaringan & Keamanan, IP Elastis.
3. Pilih Alokasi alamat IP elastis.
4. (Opsional) Ketika Anda mengalokasikan alamat IP elastis (EIP), Anda memilih grup batas jaringan untuk mengalokasikan EIP. Grup perbatasan jaringan adalah kumpulan Zona Ketersediaan (AZ), Zona Lokal, atau Zona Panjang Gelombang yang mengiklan AWS kan alamat IP publik. Zona Lokal dan Zona Panjang Gelombang mungkin memiliki grup perbatasan jaringan yang berbeda dari AZ di Wilayah untuk memastikan latensi minimum atau jarak fisik antara AWS jaringan dan pelanggan yang mengakses sumber daya di Zona ini.

   > **Important**
   >
   > Anda harus mengalokasikan EIP dalam grup perbatasan jaringan yang sama dengan AWS sumber daya yang akan dikaitkan dengan EIP. EIP dalam satu grup perbatasan jaringan hanya dapat diiklankan di zona dalam grup perbatasan jaringan tersebut dan tidak di zona lain yang diwakili oleh grup perbatasan jaringan lainnya.

   Jika Anda mengaktifkan Local Zones atau Wavelength Zone (untuk informasi selengkapnya, lihat Mengaktifkan Local Zones atau Mengaktifkan Wavelength Zone), Anda dapat memilih grup perbatasan jaringan untuk AZ, Local Zones, atau Wavelength Zone. Pilih grup perbatasan jaringan dengan hati-hati karena EIP dan sumber daya AWS yang terkait dengannya harus berada di grup perbatasan jaringan yang sama. Anda dapat menggunakan konsol EC2 untuk melihat grup perbatasan jaringan tempat Zona Ketersediaan, Zona Lokal, atau Zona Panjang Gelombang Anda berada. Biasanya, semua Zona Ketersediaan di Wilayah milik grup perbatasan jaringan yang sama, sedangkan Local Zones atau Wavelength Zone milik grup perbatasan jaringan mereka sendiri yang terpisah.

Jika Anda tidak mengaktifkan Local Zones atau Wavelength Zone, saat Anda mengalokasikan EIP, grup perbatasan jaringan yang mewakili semua AZ untuk Wilayah tersebut (seperti `us-west-2`) telah ditentukan sebelumnya untuk Anda dan Anda tidak dapat mengubahnya. Ini berarti bahwa EIP yang Anda alokasikan ke grup perbatasan jaringan ini akan diiklankan di semua AZ di Wilayah tempat Anda berada.

5. Untuk Kumpulan alamat IPv4 publik, pilih salah satu dari yang berikut:

   - Kumpulan alamat IPv4 Amazon—Jika Anda menginginkan alamat IPv4 dialokasikan dari kumpulan alamat IPv4 Amazon.
   - Alamat IPv4 publik yang Anda bawa ke AWS akun Anda —Jika Anda ingin mengalokasikan alamat IPv4 publik yang tidak bersebelahan (non-sekuensial) dari kumpulan alamat IP yang telah Anda bawa ke akun Anda. AWS Opsi ini dinonaktifkan jika Anda tidak memiliki kumpulan alamat IP. Untuk informasi selengkapnya tentang membawa rentang alamat IP Anda sendiri ke AWS akun Anda, lihatBawa alamat IP Anda sendiri (BYOIP) ke Amazon EC2.
   - Kumpulan alamat IPv4 milik pelanggan —Jika Anda ingin mengalokasikan alamat IPv4 dari kumpulan yang dibuat dari jaringan lokal Anda untuk digunakan dengan Outpost. AWS Opsi ini dinonaktifkan jika Anda tidak memiliki AWS Outpost.
   - Alokasikan menggunakan kumpulan IPv4 IPAM: Jika Anda ingin mengalokasikan alamat IP Elastic berurutan dari blok IPv4 publik yang berdekatan di kumpulan IPAM. Mengalokasikan alamat IP Elastic berurutan dapat secara signifikan mengurangi overhead manajemen untuk daftar kontrol akses keamanan dan menyederhanakan alokasi dan pelacakan alamat IP untuk penskalaan AWS perusahaan. Untuk informasi selengkapnya, lihat Meng alokasikan alamat IP Elastis berurutan dari kumpulan IPAM di Panduan Pengguna IPAM Amazon VPC.

6. (Opsional) Untuk menambahkan tag, pilih Tambahkan tag baru dan masukkan kunci tag dan nilai tag.

AWS CLI

Untuk mengalokasikan alamat Elastic IP

Gunakan perintah AWS CLI allocate-address.

Dalam contoh berikut, Amazon EC2 memilih alamat dari kumpulan alamat Amazon.

```
aws ec2 allocate-address
```

Dalam contoh berikut, Amazon EC2 memilih alamat dari kumpulan tertentu yang Anda bawa ke AWS menggunakan BYOIP.

```
aws ec2 allocate-address \
    --public-ipv4-pool ipv4pool-ec2-012345abcdef67890
```

Contoh berikut menentukan alamat dari kumpulan IPAM IPv4 yang ditentukan.

```
aws ec2 allocate-address \
    --ipam-pool-id ipam-pool-1234567890abcdef0 \
    --address 192.0.2.0
```

PowerShell

Untuk mengalokasikan Alamat IP elastis

Gunakan New-EC2Address cmdlet.

Dalam contoh berikut, Amazon EC2 memilih alamat dari kumpulan alamat Amazon.

```
New-EC2Address
```

Dalam contoh berikut, Amazon EC2 memilih alamat dari kumpulan tertentu yang Anda bawa ke AWS menggunakan BYOIP.

```
New-EC2Address `
    -PublicIpv4Pool ipv4pool-ec2-012345abcdef67890
```

Contoh berikut menentukan alamat dari kumpulan IPAM IPv4 yang ditentukan.

```
New-EC2Address `
    -IpamPoolId ipam-pool-1234567890abcdef0 `
    -Address 192.0.2.0
```

## Kaitkan sebuah alamat IP Elastis

Jika Anda mengaitkan sebuah alamat IP Elastis dengan instans Anda untuk mengaktifkan komunikasi dengan internet, Anda juga harus memastikan bahwa instans Anda berada dalam subnet publik.

Untuk informasi selengkapnya, lihat Meng aktifkan akses internet menggunakan gateway internet di Panduan Pengguna Amazon VPC.

Console

Untuk mengaitkan alamat IP Elastis dengan sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih IP Elastis.
3. Pilih alamat IP Elastis untuk dikaitkan dan pilih Tindakan, Kaitkan alamat IP Elastis.
4. Untuk Tipe sumber daya, pilih Instans.
5. Untuk instans, pilih instans yang akan dikaitkan dengan alamat IP Elastis. Anda juga dapat memasukkan teks untuk mencari instans tertentu.
6. (Opsional) Untuk Alamat IP privat, tentukan IT alamat IP privat yang akan dikaitkan dengan alamat IP Elastis.
7. Pilih Kaitkan.

Untuk mengaitkan alamat IP Elastis dengan antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih IP Elastis.
3. Pilih alamat IP Elastis untuk dikaitkan dan pilih Tindakan, Kaitkan alamat IP Elastis.
4. Untuk Tipe sumber daya, pilih Antarmuka jaringan.
5. Untuk Antarmuka jaringan, pilih antarmuka jaringan yang akan dikaitkan dengan alamat IP Elastis. Anda juga dapat memasukkan teks untuk mencari antarmuka jaringan spesifik.
6. (Opsional) Untuk Alamat IP privat, tentukan IT alamat IP privat yang akan dikaitkan dengan alamat IP Elastis.
7. Pilih Kaitkan.

AWS CLI

Untuk mengaitkan alamat IP Elastis

Gunakan perintah associate-address AWS CLI .

```
aws ec2 associate-address \
```

```
    --instance-id i-0b263919b6498b123 \
    --allocation-id eipalloc-64d5890a
```

PowerShell

Untuk mengaitkan alamat IP Elastis

Gunakan Register-EC2Address cmdlet.

```
Register-EC2Address `
    -InstanceId i-0b263919b6498b123 `
    -AllocationId eipalloc-64d5890a
```

## Pisahkan alamat IP Elastis

: Untuk melepaskan pengaitan alamat IP Elastis dari instans atau antarmuka jaringan. Setelah Anda memutuskan alamat IP Elastic, Anda dapat mengaitkannya dengan sumber daya lain.

Console

Untuk memisahkan dan mengaitkan alamat IP Elastis

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih IP Elastis.
3. Pilih alamat IP Elastis yang akan dipisahkan, pilih Tindakan, Pisahkan alamat IP Elastis.
4. Pilih Pisahkan.

AWS CLI

Untuk memisahkan alamat IP Elastis

Gunakan perintah disassociate-address AWS CLI .

```
aws ec2 disassociate-address --association-id eipassoc-12345678
```

PowerShell

Untuk memisahkan alamat IP Elastis

Gunakan Unregister-EC2Address cmdlet.

```
Unregister-EC2Address -AssociationId eipassoc-12345678
```

## Mentransfer alamat IP elastis antara Akun AWS

Anda dapat mentransfer alamat IP Elastic dari satu Akun AWS ke yang lain. Ini dapat membantu dalam situasi berikut:

- Pemulihan bencana — Memetakan ulang alamat IP dengan cepat untuk beban kerja internet yang menghadap publik selama peristiwa darurat.
- Restrukturisasi organisasi — Pindahkan beban kerja dengan cepat dari satu Akun AWS ke yang lain. Transfer alamat menghindari kebutuhan untuk menunggu alamat IP Elastic baru diizinkan oleh grup keamanan dan ACL jaringan Anda.
- Administrasi keamanan terpusat — Gunakan akun AWS keamanan terpusat untuk melacak dan mentransfer alamat IP Elastic yang telah diperiksa untuk kepatuhan keamanan.

Harga

Tidak ada biaya untuk mentransfer alamat IP Elastis.

Tugas

- Aktifkan transfer alamat IP Elastis
- Menerima alamat IP Elastis yang ditransfer
- Nonaktifkan transfer alamat IP Elastis

### Aktifkan transfer alamat IP Elastis

Bagian ini menjelaskan cara menerima alamat IP Elastis yang ditransfer. Perhatikan batasan berikut yang terkait dengan mengaktifkan alamat IP Elastis untuk transfer:

- Anda dapat mentransfer alamat IP Elastic dari apa pun Akun AWS (akun sumber) ke yang lain Akun AWS dalam Wilayah AWS (akun transfer) yang sama. Anda tidak dapat mentransfer alamat IP Elastic ke Wilayah yang berbeda.
- Saat Anda mentransfer alamat IP Elastis, ada jabat tangan dua langkah di antara. Akun AWS Ketika akun sumber memulai transfer, akun transfer memiliki tujuh hari untuk menerima transfer alamat IP Elastis. Selama tujuh hari tersebut, akun sumber dapat melihat transfer yang tertunda

(misalnya di AWS konsol atau dengan menggunakan perintah describe-address- transfer). Setelah tujuh hari, transfer berakhir dan kepemilikan alamat IP Elastis kembali ke akun sumber.

- Transfer yang diterima terlihat ke akun sumber (misalnya di AWS konsol atau dengan menggunakan perintah describe-address-transfer) selama 14 hari setelah transfer diterima.
- AWS tidak memberi tahu akun transfer tentang permintaan transfer alamat IP Elastic yang tertunda. Pemilik akun sumber harus memberi tahu pemilik akun transfer bahwa ada permintaan transfer alamat IP Elastis yang harus mereka terima.
- Tanda apa pun yang terkait dengan alamat IP Elastis yang ditransfer diatur ulang saat transfer selesai.
- Anda tidak dapat mentransfer alamat IP Elastic yang dialokasikan dari kumpulan alamat IPv4 publik yang Anda Akun AWS bawa ke kumpulan alamat Bring Your Own IP (BYOIP).
- Anda tidak dapat mentransfer alamat IP Elastic yang dialokasikan dari Amazon-provided kumpulan IPv4 Amazon VPC IP Address Manager (IPAM) publik yang berdekatan. Sebagai gantinya, IPAM memungkinkan Anda untuk berbagi kumpulan IPAM di seluruh AWS akun dengan mengintegrasikan IPAM dengan AWS Organisasi dan menggunakan. AWS RAM Untuk informasi selengkapnya, lihat Meng alokasikan alamat IP Elastis berurutan dari kumpulan IPAM di Panduan Pengguna IPAM Amazon VPC.
- Jika Anda mencoba mentransfer alamat IP Elastis yang memiliki catatan DNS terbalik yang terkait dengannya, Anda dapat memulai proses transfer, tetapi akun transfer tidak akan dapat menerima transfer sampai catatan DNS terkait dihapus.
- Jika Anda telah mengaktifkan dan mengonfigurasi AWS Outposts, Anda mungkin telah mengalokasikan alamat IP Elastic dari kumpulan alamat IP milik pelanggan (CoIP). Anda tidak dapat mentransfer alamat IP Elastis yang dialokasikan dari CoIP. Namun, Anda dapat menggunakan AWS RAM untuk berbagi CoIP dengan akun lain. Untuk informasi selengkapnya, lihat alamat Customer-owned IP di Panduan AWS Outposts Pengguna.
- Anda dapat menggunakan Amazon VPC IPAM untuk melacak transfer alamat IP Elastis ke akun di organisasi AWS Organizations. Untuk informasi selengkapnya, lihat Lihat riwayat alamat IP. Jika alamat IP Elastic ditransfer ke Akun AWS luar organisasi, riwayat audit IPAM dari alamat IP Elastic hilang.

Langkah-langkah ini harus diselesaikan oleh akun sumber.

Console

Untuk mengaktifkan transfer alamat IP Elastis

1. Pastikan Anda menggunakan AWS akun sumber.
2. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
3. Di panel navigasi, pilih IP Elastis.
4. Pilih satu atau beberapa alamat IP elastis untuk mengaktifkan transfer dan pilih Tindakan, Aktifkan transfer.
5. Jika Anda mentransfer beberapa alamat IP Elastis, Anda akan melihat opsi Tipe transfer. Pilih salah satu opsi berikut:
   - Pilih Akun tunggal jika Anda mentransfer alamat IP Elastic ke satu AWS akun.
   - Pilih Beberapa akun jika Anda mentransfer alamat IP Elastic ke beberapa AWS akun.
6. Di bawah Transfer ID akun, masukkan ID akun AWS yang ingin Anda transfer alamat IP Elastis.
7. Konfirmasikan transfer dengan memasukkan **`enable`** dalam kotak teks.
8. Pilih Kirim.
9. Untuk menerima transfer, lihat Menerima alamat IP Elastis yang ditransfer. Untuk menonaktifkan transfer, lihat Nonaktifkan transfer alamat IP Elastis.

AWS CLI

Untuk mengaktifkan transfer alamat IP Elastis

Gunakan perintah enable-address-transfer.

```
aws ec2 enable-address-transfer \
    --allocation-id eipalloc-09ad461b0d03f6aaf \
    --transfer-account-id 123456789012
```

PowerShell

Untuk mengaktifkan transfer alamat IP Elastis

Gunakan Enable-EC2AddressTransfer cmdlet.

```
Enable-EC2AddressTransfer `
    -AllocationId eipalloc-09ad461b0d03f6aaf `
    -TransferAccountId 123456789012
```

## Menerima alamat IP Elastis yang ditransfer

Bagian ini menjelaskan cara menerima alamat IP Elastis yang ditransfer.

Saat Anda mentransfer alamat IP Elastis, ada jabat tangan dua langkah di antara. Akun AWS Ketika akun sumber memulai transfer, akun transfer memiliki tujuh hari untuk menerima transfer alamat IP Elastis. Selama tujuh hari tersebut, akun sumber dapat melihat transfer yang tertunda (misalnya di AWS konsol atau dengan menggunakan perintah describe-address- transfer). Setelah tujuh hari, transfer berakhir dan kepemilikan alamat IP Elastis kembali ke akun sumber.

Saat menerima transfer, perhatikan pengecualian berikut yang mungkin terjadi dan cara mengatasinya:

- AddressLimitExceeded: Jika akun transfer Anda telah melebihi kuota alamat IP Elastic, akun sumber dapat mengaktifkan transfer alamat IP Elastic, tetapi pengecualian ini terjadi ketika akun transfer mencoba menerima transfer. Secara default, semua AWS akun dibatasi hingga 5 alamat IP elastis per Wilayah. Lihat Kuota alamat IP Elastis untuk instruksi tentang meningkatkan batas.
- InvalidTransfer.AddressCustomPtrSet: Jika Anda atau seseorang di organisasi Anda telah mengonfigurasi alamat IP Elastic yang Anda coba transfer untuk menggunakan pencarian DNS terbalik, akun sumber dapat mengaktifkan transfer untuk alamat IP Elastic, tetapi pengecualian ini terjadi ketika akun transfer mencoba menerima transfer. Untuk mengatasi masalah ini, akun sumber harus menghapus catatan DNS untuk alamat IP Elastis. Untuk informasi selengkapnya, lihat Buat catatan DNS terbalik untuk email di Amazon EC2.
- InvalidTransfer.AddressAssociated: Jika alamat IP Elastic dikaitkan dengan instans ENI atau EC2, akun sumber dapat mengaktifkan transfer untuk alamat IP Elastic, tetapi pengecualian ini terjadi ketika akun transfer mencoba menerima transfer. Untuk mengatasi masalah ini, akun sumber harus memisahkan alamat IP Elastis. Untuk informasi selengkapnya, lihat Pisahkan alamat IP Elastis.

Untuk pengecualian lainnya, hubungi Dukungan.

Langkah-langkah ini harus diselesaikan oleh akun transfer.

Console

Untuk menerima transfer alamat IP Elastis

1. Pastikan Anda menggunakan akun transfer.
2. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
3. Di panel navigasi, pilih IP Elastis.
4. Pilih Tindakan, Terima transfer.
5. Tidak ada tanda yang terkait dengan alamat IP Elastis yang ditransfer dengan alamat IP Elastis saat Anda menerima transfer. Jika Anda ingin menentukan tanda Nama untuk alamat IP Elastis yang Anda terima, pilih Buat tanda dengan kunci 'Nama' dan nilai yang Anda tentukan.
6. Masukkan alamat IP Elastis yang ingin Anda transfer.
7. Jika Anda menerima beberapa alamat IP Elastis yang ditransfer, pilih Tambah alamat untuk memasukkan alamat IP Elastis tambahan.
8. Pilih Kirim.

AWS CLI

Untuk menerima transfer alamat IP Elastis

Gunakan perintah accept-address-transfer.

```
aws ec2 accept-address-transfer --address 100.21.184.216
```

PowerShell

Untuk menerima transfer alamat IP Elastis

Gunakan Approve-EC2AddressTransfer cmdlet.

```
Approve-EC2AddressTransfer -Address 100.21.184.216
```

## Nonaktifkan transfer alamat IP Elastis

Bagian ini menjelaskan cara menonaktifkan transfer IP Elastis setelah transfer diaktifkan.

Langkah-langkah ini harus diselesaikan oleh akun sumber yang mengaktifkan transfer.

Console

Untuk menonaktifkan transfer alamat IP Elastis

1. Pastikan Anda menggunakan sumbernya Akun AWS.
2. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
3. Di panel navigasi, pilih IP Elastis.
4. Dalam daftar sumber daya IP Elastis, pastikan properti Anda diaktifkan yang menampilkan status Transfer kolom.
5. Pilih satu atau beberapa alamat IP elastis yang memiliki status Transfer Tertunda, dan pilih Tindakan, Nonaktifkan transfer.
6. Konfirmasikan dengan memasukkan **`disable`** di kotak teks.
7. Pilih Kirim.

AWS CLI

Untuk menonaktifkan transfer alamat IP Elastis

Gunakan perintah disable-address-transfer.

```
aws ec2 disable-address-transfer --allocation-id eipalloc-09ad461b0d03f6aaf
```

PowerShell

Untuk menonaktifkan transfer alamat IP Elastis

Gunakan Disable-EC2AddressTransfer cmdlet.

```
Disable-EC2AddressTransfer -AllocationId eipalloc-09ad461b0d03f6aaf
```

## Merilis alamat IP Elastis

Jika Anda tidak lagi memerlukan alamat IP Elastis, kami menyarankan Anda melepaskannya. Alamat IP Elastic yang akan dirilis saat ini tidak boleh dikaitkan dengan AWS sumber daya.

Console

Untuk merilis alamat IP Elastis

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih IP Elastis.
3. Pilih alamat IP Elastis yang akan dilepas dan pilih Tindakan, Lepas alamat IP Elastis.
4. Pilih Lepas.

AWS CLI

Untuk merilis alamat IP Elastis

Gunakan perintah release-address AWS CLI .

```
aws ec2 release-address --allocation-id eipalloc-64d5890a
```

PowerShell

Untuk merilis alamat IP Elastis

Gunakan Remove-EC2Address cmdlet.

```
Remove-EC2Address -AllocationId eipalloc-64d5890a
```

Setelah Anda melepaskan alamat IP Elastic Anda, Anda mungkin dapat memulihkan. Aturan-aturan berikut berlaku:

- Anda tidak dapat memulihkan alamat IP Elastic jika telah dialokasikan ke AWS akun lain, atau jika itu akan mengakibatkan Anda melebihi batas alamat IP Elastic Anda.
- Anda tidak dapat memulihkan tag yang terkait dengan alamat IP Elastic.

AWS CLI

Untuk memulihkan alamat IP Elastis

Gunakan perintah allocate-address.

```
aws ec2 allocate-address \
    --domain vpc \
    --address 203.0.113.3
```

PowerShell

Untuk memulihkan alamat IP Elastis

Gunakan New-EC2Address cmdlet.

```
New-EC2Address `
    -Address 203.0.113.3 `
    -Domain vpc `
    -Region us-east-1
```

## Buat catatan DNS terbalik untuk email di Amazon EC2

Jika Anda ingin mengirim email ke pihak ketiga dari instans EC2, sebaiknya Anda menyediakan satu atau lebih alamat IP Elastic dan menetapkan catatan DNS terbalik statis ke alamat IP Elastic yang Anda gunakan untuk mengirim email. Ini dapat membantu Anda menghindari email Anda ditandai sebagai spam oleh beberapa organisasi anti-spam. AWS bekerja dengan ISP dan organisasi anti-spam internet untuk mengurangi kemungkinan email Anda yang dikirim dari alamat ini akan ditandai sebagai spam.

Pertimbangan-pertimbangan

- Sebelum membuat catatan DNS terbalik, Anda harus menetapkan catatan DNS maju yang sesuai (tipe rekaman A) yang diselesaikan ke alamat IP Elastic di akun Anda. Sebagai praktik terbaik, sertakan semua alamat IP Elastic yang ingin Anda buat catatan DNS terbalik dalam catatan DNS maju.
- Jika catatan DNS terbalik dikaitkan dengan alamat IP Elastis, alamat IP Elastis terkunci ke akun Anda dan tidak dapat dikeluarkan dari akun Anda hingga rekaman dihapus.
- Jika Anda menghubungi Dukungan untuk mengatur DNS terbalik untuk alamat IP Elastic, Anda dapat menghapus DNS terbalik, tetapi Anda tidak dapat melepaskan alamat IP Elastic karena dikunci oleh Dukungan. Untuk membuka kunci alamat IP Elastis, hubungi AWS Dukungan. Setelah alamat IP Elastic dibuka, Anda dapat melepaskannya.
- [AWS GovCloud (US) Region] Anda tidak dapat membuat catatan DNS terbalik. AWS harus menetapkan catatan DNS terbalik statis untuk Anda. Buka kasus dukungan untuk menghapus

DNS terbalik dan batasan pengiriman email. Anda harus memberikan alamat IP Elastic Anda dan membalikkan catatan DNS.

## Membuat catatan DNS terbalik

Anda dapat membuat catatan DNS terbalik untuk alamat IP Elastic Anda sebagai berikut.

### Console

Untuk membuat catatan DNS terbalik

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih IP Elastis.
3. Pilih alamat IP Elastis dan pilih Tindakan, Memperbarui DNS terbalik.
4. Untuk Membalikkan nama domain DNS, masukkan nama domain.
5. Masukkan **`update`** untuk mengonfirmasi.
6. Pilih Perbarui.

### AWS CLI

Untuk membuat catatan DNS terbalik

Gunakan perintah modify-address-attribute.

```
aws ec2 modify-address-attribute \
    --allocation-id eipalloc-abcdef01234567890 \
    --domain-name example.com
```

Berikut ini adalah output contoh.

```
{
    "Addresses": [
        {
            "PublicIp": "192.0.2.0",
            "AllocationId": "eipalloc-abcdef01234567890",
            "PtrRecord": "example.net.",
            "PtrRecordUpdate": {
                "Value": "example.com.",
```

```
                "Status": "PENDING"
            }
        }
    ]
}
```

PowerShell

Untuk membuat catatan DNS terbalik

Gunakan Edit-EC2AddressAttribute cmdlet.

```
Edit-EC2AddressAttribute `
    -AllocationId 'eipalloc-abcdef01234567890' `
    -DomainName 'example.com' |
Format-List `
    AllocationId, PtrRecord, PublicIp,
    @{Name='PtrRecordUpdate';Expression={$_.PtrRecordUpdate | Format-List | Out-
String}}
```

Berikut ini adalah output contoh.

```
AllocationId    : eipalloc-abcdef01234567890
PtrRecord       : example.net.
PublicIp        : 192.0.2.0
PtrRecordUpdate :
                  Reason :
                  Status : PENDING
                  Value  : example.com.
```

## Membuat catatan DNS terbalik

Anda dapat menghapus catatan DNS terbalik dari alamat IP Elastic Anda sebagai berikut.

Jika Anda menerima kesalahan berikut, Anda dapat mengirimkan Per mintaan untuk menghapus batasan pengiriman email ke Dukungan untuk bantuan.

```
The address cannot be released because it is locked to your account.
```

Console

Untuk menghapus catatan DNS terbalik

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih IP Elastis.
3. Pilih alamat IP Elastis dan pilih Tindakan, Memperbarui DNS terbalik.
4. Untuk nama domain DNS Terbalik, hapus nama domain.
5. Masukkan **`update`** untuk mengonfirmasi.
6. Pilih Perbarui.

AWS CLI

Untuk menghapus catatan DNS terbalik

Gunakan perintah reset-address-attribute.

```
aws ec2 reset-address-attribute \
    --allocation-id eipalloc-abcdef01234567890 \
    --attribute domain-name
```

Berikut ini adalah output contoh.

```
{
    "Addresses": [
        {
            "PublicIp": "192.0.2.0",
            "AllocationId": "eipalloc-abcdef01234567890",
            "PtrRecord": "example.com.",
            "PtrRecordUpdate": {
                "Value": "example.net.",
                "Status": "PENDING"
            }
        }
    ]
}
```

PowerShell

Untuk menghapus catatan DNS terbalik

Gunakan Reset-EC2AddressAttribute cmdlet.

```
Reset-EC2AddressAttribute `
    -AllocationId 'eipalloc-abcdef01234567890' `
    -Attribute domain-name |
Format-List `
    AllocationId, PtrRecord, PublicIp,
    @{Name='PtrRecordUpdate';Expression={$_.PtrRecordUpdate | Format-List | Out-
String}}
```

Berikut ini adalah output contoh.

```
AllocationId    : eipalloc-abcdef01234567890
PtrRecord       : example.com.
PublicIp        : 192.0.2.0
PtrRecordUpdate :
                  Reason :
                  Status : PENDING
                  Value  : example.net.
```

# Antarmuka jaringan elastis

Antarmuka jaringan elastis adalah komponen jaringan logis dalam VPC yang mewakili kartu jaringan virtual. Anda dapat membuat dan mengonfigurasi antarmuka jaringan dan melampirkannya ke instans yang Anda luncurkan di Zona Ketersediaan yang sama. Atribut antarmuka jaringan mengikutinya saat dilampirkan atau dilepaskan dari sebuah instans dan dilampirkan kembali ke instans lain. Saat Anda memindahkan antarmuka jaringan dari satu instance ke instance lainnya, lalu lintas jaringan dialihkan dari instance asli ke instance baru.

Perhatikan bahwa AWS sumber daya ini disebut sebagai antarmuka  jaringan  di Konsol Manajemen AWS dan Amazon EC2 API. Oleh karena itu, kami menggunakan "antarmuka jaringan" dalam dokumentasi ini daripada "antarmuka jaringan elastis". Istilah "antarmuka jaringan" dalam dokumentasi ini selalu berarti "antarmuka jaringan elastis".

Atribut antarmuka jaringan

Antarmuka jaringan dapat mencakup atribut berikut:

- Alamat IPv4 pribadi utama dari rentang alamat IPv4 subnet Anda

- Alamat IPv6 utama dari rentang alamat IPv6 subnet Anda
- Alamat IPv4 pribadi sekunder dari rentang alamat IPv4 subnet Anda
- Satu alamat IP elastis (IPv4) untuk setiap alamat IPv4 pribadi
- Satu alamat IPv4 publik
- Alamat IPv6 sekunder
- Grup keamanan
- Alamat MAC
- Bendera source/destination cek
- Deskripsi

Memantau lalu lintas

Anda dapat mengaktifkan log aliran VPC pada antarmuka jaringan Anda untuk menangkap informasi tentang lalu lintas yang menuju dan dari antarmuka jaringan. Setelah membuat log alur, Anda dapat melihat dan mengambil datanya di Amazon CloudWatch Logs. Untuk informasi selengkapnya, lihat Log Alur VPC di Panduan Pengguna Amazon VPC.

Daftar Isi


- Konsep antarmuka jaringan
- Kartu jaringan
- Alamat IP maksimum per antarmuka jaringan
- Membuat antarmuka jaringan untuk instans EC2 Anda
- Lampiran antarmuka jaringan untuk instans EC2 Anda
- Kelola alamat IP untuk antarmuka jaringan Anda
- Memodifikasi atribut antarmuka jaringan
- Beberapa antarmuka jaringan untuk instans Amazon EC2 Anda
- Requester-managed antarmuka jaringan
- Delegasi awalan untuk antarmuka jaringan Amazon EC2
- Menghapus antarmuka jaringan


## Konsep antarmuka jaringan

Berikut ini adalah konsep penting untuk dipahami saat Anda mulai menggunakan antarmuka jaringan.

Antarmuka jaringan primer

Setiap instans memiliki antarmuka jaringan default, yang disebut antarmuka jaringan primer. Anda tidak dapat melepaskan antarmuka jaringan utama dari sebuah instance.

Antarmuka jaringan sekunder

Anda dapat membuat dan melampirkan antarmuka jaringan sekunder ke instance Anda. Jumlah maksimum antarmuka jaringan bervariasi menurut jenis instance. Untuk informasi selengkapnya, lihat Alamat IP maksimum per antarmuka jaringan.

Alamat IPv4 untuk antarmuka jaringan

Saat Anda meluncurkan instance EC2 ke subnet IPv4-only atau dual stack, instans menerima alamat IP pribadi utama dari rentang alamat IPv4 subnet. Anda juga dapat menentukan alamat IPv4 privat tambahan, yang dikenal sebagai alamat IPv4 privat sekunder. Tidak seperti alamat IP privat primer, alamat IP privat sekunder dapat ditetapkan ulang dari satu instans ke instans lainnya.

Alamat IPv4 publik untuk antarmuka jaringan

Semua subnet memiliki atribut yang dapat dimodifikasi yang menentukan apakah antarmuka jaringan yang dibuat di subnet itu (dan oleh karena itu instance yang diluncurkan ke subnet itu) diberi alamat IPv4 publik. Untuk informasi selengkapnya, lihat Pengaturan subnet di Panduan Pengguna Amazon VPC. Saat Anda meluncurkan instance, alamat IP ditetapkan ke antarmuka jaringan utama. Jika Anda menentukan antarmuka jaringan yang ada sebagai antarmuka jaringan utama saat Anda meluncurkan instance, alamat IPv4 publik ditentukan oleh antarmuka jaringan ini.

Saat Anda membuat antarmuka jaringan, ini mewarisi IPv4 publik yang mengalamatkan atribut dari subnet. Jika nanti Anda memodifikasi atribut pengalamatan IPv4 publik dari subnet, antarmuka jaringan menyimpan pengaturan yang berlaku saat dibuat.

Kami merilis alamat IP publik ketika instance dihentikan, hibernasi, atau dihentikan. Kami menetapkan alamat IP publik baru saat Anda memulai instance yang dihentikan atau hibernasi, kecuali jika memiliki antarmuka jaringan sekunder atau alamat IPv4 pribadi sekunder yang terkait dengan alamat IP Elastic.

Alamat IPv6 untuk antarmuka jaringan

Jika Anda mengaitkan blok IPv6 CIDR dengan VPC dan subnet Anda, Anda dapat menetapkan alamat IPv6 dari rentang subnet ke antarmuka jaringan. Setiap alamat IPv6 dapat diberikan ke satu antarmuka jaringan.

Semua subnet memiliki atribut yang dapat dimodifikasi yang menentukan apakah antarmuka jaringan yang dibuat di subnet tersebut (dan oleh karena itu instans yang diluncurkan ke subnet tersebut) secara otomatis diberi alamat IPv6 dari kisaran subnet. Saat Anda meluncurkan instance, alamat IPv6 ditetapkan ke antarmuka jaringan utama.

Alamat IP elastis untuk antarmuka jaringan

Anda dapat mengaitkan alamat IP Elastic dengan salah satu alamat IPv4 pribadi untuk antarmuka jaringan. Anda dapat mengaitkan satu alamat IP Elastis dengan setiap alamat IPv4 privat. Jika Anda melepaskan alamat IP Elastic dari antarmuka jaringan, Anda dapat melepaskannya atau mengaitkannya dengan instance yang berbeda.

Perilaku pemutusan hubungan kerja

Anda dapat menyetel perilaku terminasi untuk antarmuka jaringan yang dilampirkan ke sebuah instans. Anda dapat menentukan apakah antarmuka jaringan harus dihapus secara otomatis saat Anda menghentikan instans yang dilampirkan.

Source/destination memeriksa

Anda dapat mengaktifkan atau menonaktifkan source/destination pemeriksaan, yang memastikan bahwa instance adalah sumber atau tujuan dari lalu lintas apa pun yang diterimanya. Source/destination pemeriksaan diaktifkan secara default. Anda harus menonaktifkan source/destination pemeriksaan jika instance menjalankan layanan seperti terjemahan alamat jaringan, routing, atau firewall.

Requester-managed antarmuka jaringan

Antarmuka jaringan ini dibuat dan dikelola oleh Layanan AWS untuk memungkinkan Anda menggunakan beberapa sumber daya dan layanan. Anda tidak dapat mengelola antarmuka jaringan ini sendiri. Untuk informasi selengkapnya, lihat Requester-managed antarmuka jaringan.

Delegasi awalan

Awalan adalah rentang IPv4 atau IPv6 CIDR pribadi yang dicadangkan yang Anda alokasikan untuk penugasan otomatis atau manual ke antarmuka jaringan yang terkait dengan instance. Dengan menggunakan Prefiks yang Didelegasikan, Anda dapat meluncurkan layanan lebih cepat dengan menetapkan berbagai alamat IP sebagai prefiks tunggal.

Antarmuka jaringan yang dikelola

Antar muka jaringan ter kelola dikelola oleh penyedia layanan, seperti Mode Otomatis Amazon EKS. Anda tidak dapat secara langsung mengubah pengaturan antarmuka jaringan terkelola.

Antarmuka jaringan terkelola diidentifikasi oleh nilai sebenarnya di bidang Terkelola. Untuk informasi selengkapnya, lihat Instans terkelola Amazon EC2.

## Kartu jaringan

Sebagian besar jenis instance mendukung satu kartu jaringan. Jenis instans yang mendukung beberapa kartu jaringan memberikan kinerja jaringan yang lebih tinggi, termasuk kemampuan bandwidth di atas 100 Gbps dan peningkatan kinerja kecepatan paket. Saat Anda melampirkan antarmuka jaringan ke instance yang mendukung beberapa kartu jaringan, Anda dapat memilih kartu jaringan untuk antarmuka jaringan. Antarmuka jaringan primer harus ditetapkan ke indeks kartu jaringan 0.

EFA dan antarmuka EFA-only jaringan dihitung sebagai antarmuka jaringan. Anda hanya dapat menetapkan satu EFA atau antarmuka EFA-only jaringan per kartu jaringan. Antarmuka jaringan utama tidak bisa menjadi antarmuka EFA-only jaringan.

Jenis instance berikut mendukung beberapa kartu jaringan. Untuk informasi tentang jumlah antarmuka jaringan yang didukung oleh jenis instance, lihatAlamat IP maksimum per antarmuka jaringan.

| Tipe instans | Jumlah kartu jaringan |
|---|---|
| c6in.32xlarge | 2 |
| c6in.metal | 2 |
| c8gb.48xlarge | 2 |
| c8gb.metal-48xl | 2 |
| c8gn.48xlarge | 2 |
| c8gn.metal-48xl | 2 |
| c8in.96xlarge | 2 |
| c8in.metal-96xl | 2 |
| c8ib.96xlarge | 2 |

| Tipe instans | Jumlah kartu jaringan |
| --- | --- |
| c8ib.metal-96xl | 2 |
| dl1.24xlarge | 4 |
| g6e.24xlarge | 2 |
| g6e.48xlarge | 4 |
| g7.48xlarge | 2 |
| g7e.24xlarge | 2 |
| g7e.48xlarge | 4 |
| hpc6id.32xlarge | 2 |
| hpc7a.12xlarge | 2 |
| hpc7a.24xlarge | 2 |
| hpc7a.48xlarge | 2 |
| hpc7a.96xlarge | 2 |
| hpc8a.96xlarge | 2 |
| i7ie.48xlarge | 2 |
| i7ie.metal-48xl | 2 |
| i8ge.48xlarge | 2 |
| i8ge.metal-48xl | 2 |
| m6idn.32xlarge | 2 |
| m6idn.metal | 2 |
| m6in.32xlarge | 2 |

| Tipe instans | Jumlah kartu jaringan |
|---|---|
| m6in.metal | 2 |
| m8gb.48xlarge | 2 |
| m8gb.metal-48xl | 2 |
| m8gn.48xlarge | 2 |
| m8gn.metal-48xl | 2 |
| m8in.96xlarge | 2 |
| m8in.metal-96xl | 2 |
| m8idn.96xlarge | 2 |
| m8idn.metal-96xl | 2 |
| m8ib.96xlarge | 2 |
| m8ib.metal-96xl | 2 |
| m8idb.96xlarge | 2 |
| m8idb.metal-96xl | 2 |
| p4d.24xlarge | 4 |
| p4de.24xlarge | 4 |
| p5.48xlarge | 32 |
| p5e.48xlarge | 32 |
| p5en.48xlarge | 16 |
| p6-b200.48xlarge | 8 |
| p6-b300.48xlarge | 17 |

| Tipe instans | Jumlah kartu jaringan |
|---|---|
| p6e-gb200.36xlarge | 17 |
| r8gb.48xlarge | 2 |
| r8gb.metal-48xl | 2 |
| r8gn.48xlarge | 2 |
| r8gn.metal-48xl | 2 |
| r8in.96xlarge | 2 |
| r8in.metal-96xl | 2 |
| r8idn.96xlarge | 2 |
| r8idn.metal-96xl | 2 |
| r8ib.96xlarge | 2 |
| r8ib.metal-96xl | 2 |
| r8idb.96xlarge | 2 |
| r8idb.metal-96xl | 2 |
| r6idn.32xlarge | 2 |
| r6idn.metal | 2 |
| r6in.32xlarge | 2 |
| r6in.metal | 2 |
| trn1.32xlarge | 8 |
| trn1n.32xlarge | 16 |
| trn2.48xlarge | 16 |

| Tipe instans | Jumlah kartu jaringan |
| --- | --- |
| trn2u.48xlarge | 16 |
| u7in-16tb.224xlarge | 2 |
| u7in-24tb.224xlarge | 2 |
| u7in-32tb.224xlarge | 2 |
| u7inh-32tb.480xlarge | 2 |

## Alamat IP maksimum per antarmuka jaringan

Setiap jenis instance mendukung jumlah maksimum antarmuka jaringan, jumlah maksimum alamat IPv4 pribadi per antarmuka jaringan, dan jumlah maksimum alamat IPv6 per antarmuka jaringan. Batas untuk alamat IPv6 terpisah dari batas alamat IPv4 privat per antarmuka jaringan. Perhatikan bahwa semua jenis instans mendukung pengalamatan IPv6 kecuali untuk yang berikut: C1, M1, M2, M3, dan T1.

Antarmuka jaringan yang tersedia

Panduan Jenis Instans  Amazon EC2  menyediakan informasi tentang antarmuka jaringan yang tersedia untuk setiap jenis instans. Untuk informasi selengkapnya, lihat berikut ini:

- Spesifikasi jaringan — Tujuan umum
- Spesifikasi jaringan - Komputasi dioptimalkan
- Spesifikasi jaringan - Memori dioptimalkan
- Spesifikasi jaringan - Penyimpanan dioptimalkan
- Spesifikasi jaringan - Komputasi yang dipercepat
- Spesifikasi jaringan — High-performance komputasi
- Spesifikasi jaringan — Generasi sebelumnya

### Console

Untuk mengambil antarmuka jaringan maksimum

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Tipe Instans.
3. Tambahkan filter untuk menentukan jenis instance (tipe instans = c5.12xlarge) atau keluarga instance (keluarga instans = c5).
4. (Opsional) Pilih ikon Preferensi dan kemudian aktifkan Jumlah maksimum antarmuka jaringan. Kolom ini menunjukkan jumlah maksimum antarmuka jaringan untuk setiap jenis instance.
5. (Opsional) Pilih jenis instans. Pada tab Jaringan, temukan Jumlah maksimum antarmuka jaringan.

### AWS CLI

Untuk mengambil antarmuka jaringan maksimum

Anda dapat menggunakan perintah describe-instance-types untuk menampilkan informasi tentang jenis instance, seperti antarmuka jaringan yang didukung dan alamat IP per antarmuka. Contoh berikut menampilkan informasi ini untuk semua instans C8i.

```
{ echo -e "InstanceType\tMaximumNetworkInterfaces\tIpv4AddressesPerInterface"; \
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=c8i.*" \
    --query 'InstanceTypes[*].[InstanceType, NetworkInfo.MaximumNetworkInterfaces,
 NetworkInfo.Ipv4AddressesPerInterface]' \
    --output text | sort -k2 -n; } | column -t
```

Berikut ini adalah output contoh.

```
InstanceType    MaximumNetworkInterfaces  Ipv4AddressesPerInterface
c8i.large       3                         20
c8i.2xlarge     4                         30
c8i.xlarge      4                         30
c8i.4xlarge     8                         50
c8i.8xlarge     10                        50
c8i.12xlarge    12                        50
c8i.16xlarge    16                        64
```

```
c8i.24xlarge    16                          64
c8i.32xlarge    24                          64
c8i.48xlarge    24                          64
c8i.96xlarge    24                          64
c8i.metal-48xl  24                          64
c8i.metal-96xl  24                          64
```

PowerShell

Untuk mengambil antarmuka jaringan maksimum

Anda dapat menggunakan Get-EC2InstanceType PowerShell perintah untuk menampilkan informasi tentang jenis instance, seperti antarmuka jaringan yang didukung dan alamat IP per antarmuka. Contoh berikut menampilkan informasi ini untuk semua instans C8i.

```
Get-EC2InstanceType -Filter @{Name="instance-type"; Values="c8i.*"} |
Select-Object `
    InstanceType,
    @{Name='MaximumNetworkInterfaces';
 Expression={$_.NetworkInfo.MaximumNetworkInterfaces}},
    @{Name='Ipv4AddressesPerInterface';
 Expression={$_.NetworkInfo.Ipv4AddressesPerInterface}} |
Sort-Object MaximumNetworkInterfaces |
Format-Table -AutoSize
```

Berikut ini adalah output contoh.

```
InstanceType   MaximumNetworkInterfaces Ipv4AddressesPerInterface
------------   ------------------------ -------------------------
c8i.large                             3                        20
c8i.xlarge                            4                        30
c8i.2xlarge                           4                        30
c8i.4xlarge                           8                        50
c8i.8xlarge                          10                        50
c8i.12xlarge                         12                        50
c8i.24xlarge                         16                        64
c8i.16xlarge                         16                        64
c8i.96xlarge                         24                        64
c8i.48xlarge                         24                        64
c8i.metal-96xl                       24                        64
c8i.32xlarge                         24                        64
c8i.metal-48xl                       24                        64
```

# Membuat antarmuka jaringan untuk instans EC2 Anda

Anda dapat membuat antarmuka jaringan untuk digunakan oleh instans EC2 Anda. Saat Anda membuat antarmuka jaringan, Anda menentukan subnet yang membuatnya. Anda tidak dapat memindahkan antarmuka jaringan ke subnet lain setelah dibuat. Anda harus memasang antarmuka jaringan ke sebuah instans di Zona Ketersediaan yang sama. Anda dapat melepaskan antarmuka jaringan sekunder dari instance dan kemudian melampirkannya ke instance yang berbeda di Zona Ketersediaan yang sama. Anda tidak dapat melepaskan antarmuka jaringan utama dari sebuah instance. Untuk informasi selengkapnya, lihat the section called “Lampiran antarmuka jaringan”.

Console

Untuk membuat antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih Buat antarmuka jaringan.
4. (Opsional) Untuk Deskripsi, masukkan nama deskriptif.
5. Untuk Subnet, pilih subnet. Opsi yang tersedia dalam langkah-langkah selanjutnya berubah tergantung pada jenis subnet yang Anda pilih (IPv4-only, IPv6-only, atau dual-stack (IPv4 dan IPv6)).
6. Untuk Jenis antarmuka, pilih salah satu dari berikut ini:

   - ENA: Antarmuka jaringan berkinerja tinggi yang dirancang untuk menangani throughput tinggi dan kecepatan paket-per-detik untuk TCP/IP protokol sambil meminimalkan penggunaan CPU. Ini adalah nilai default. Untuk informasi selengkapnya tentang ENA, lihat Adaptor Jaringan Elastis.
   - EFA dengan ENA: Antarmuka jaringan yang mendukung perangkat ENA dan EFA untuk transportasi TCP/IP berbasis tradisional bersama dengan transportasi berbasis SRD. Jika Anda memilih EFA dengan ENA, instance yang Anda lampirkan harus mendukung EF A. Untuk informasi selengkapnya tentang EFA, lihat Adaptor Kain Elastis.
   - EFA-only: Antarmuka jaringan berkinerja tinggi yang dirancang untuk menangani komunikasi antar node throughput tinggi, latensi rendah untuk transportasi berbasis SRD sambil melewati tumpukan sistem operasi. Jika Anda memilih opsi ini, instance yang Anda lampirkan harus mendukung EFA. EFA-only antarmuka jaringan tidak mendukung alamat IP. Untuk informasi selengkapnya tentang EFA, lihat Adaptor Kain Elastis.

7. Untuk Alamat IPv4 privat, lakukan salah satu langkah berikut:
   - Pilih Auto-assign untuk mengizinkan Amazon EC2 memilih alamat IPv4 dari subnet.
   - Pilih Kustom dan masukkan alamat IPv4 yang Anda pilih dari subnet.
8. (Subnet dengan alamat IPv6 saja) Untuk Alamat IPv6, lakukan salah satu langkah berikut:
   - Pilih Tidak ada jika Anda tidak ingin menetapkan alamat IPv6 ke antarmuka jaringan.
   - Pilih Auto-assign untuk mengizinkan Amazon EC2 memilih alamat IPv6 dari subnet.
   - Pilih Kustom dan masukkan alamat IPv6 yang Anda pilih dari subnet.
9. (Opsional) Jika Anda membuat antarmuka jaringan di tumpukan ganda atau IPv6-only subnet, Anda memiliki opsi untuk Menetapkan IP IP v6 Utama. Ini menetapkan alamat unicast global IPv6 primer (GUA) ke antarmuka jaringan. Dengan menetapkan alamat IPv6 primer, Anda akan dapat menghindari mengganggu lalu lintas ke instans atau ENI. Pilih Aktifkan jika instance yang akan dilampirkan ENI ini bergantung pada alamat IPv6-nya yang tidak berubah. AWS akan secara otomatis menetapkan alamat IPv6 yang terkait dengan ENI yang dilampirkan ke instance Anda untuk menjadi alamat IPv6 utama. Setelah Anda mengaktifkan alamat GUA IPv6 menjadi IPv6 primer, Anda tidak dapat menonaktifkannya. Saat Anda mengaktifkan alamat IPv6 GUA menjadi IPv6 primer, IPv6 GUA pertama akan dijadikan alamat IPv6 primer sampai instans diakhiri atau antarmuka jaringan dilepas. Jika Anda memiliki beberapa alamat IPv6 yang terkait dengan ENI yang dilampirkan ke instans Anda dan Anda mengaktifkan alamat IPv6 primer, alamat IPv6 GUA pertama yang terkait dengan ENI menjadi alamat IPv6 primer.
10. (Opsional) Untuk membuat Elastic Fabric Adapter, pilih Elastic Fabric Adapter, Aktifkan.
11. (Opsional) Di bawah Pengaturan lanj utan, Anda dapat mengatur delegasi awalan IP secara opsional. Untuk informasi selengkapnya, lihat Delegasi awalan.
    - Auto-assign— AWS memilih awalan dari blok IPv4 atau IPv6 CIDR untuk subnet, dan menetapkannya ke antarmuka jaringan.
    - Kustom — Anda menentukan awalan dari blok CIDR IPv4 atau IPv6 untuk subnet, dan AWS memverifikasi bahwa awalan belum ditetapkan ke sumber daya lain sebelum menetapkannya ke antarmuka jaringan.
12. (Opsional) Di bawah Pengaturan lanjutan, untuk Batas waktu pelacakan koneksi idle, modifikasi batas waktu koneksi idle default. Untuk informasi selengkapnya, lihat Waktu habis pelacakan koneksi idle.
    - TCP menetapkan batas waktu: Batas waktu (dalam detik) untuk koneksi TCP idle dalam keadaan mapan.

- Min: 60 detik
- Maks: 432000 detik
- Default: 350 detik untuk jenis instance Nitro v6 (tidak termasuk P6e-GB200); 432000 detik untuk semua jenis instance lainnya (termasuk P6e-GB200).
- Direkomendasikan: Kurang dari 432000 detik

- Batas waktu UDP: Batas waktu (dalam detik) untuk alur UDP idle yang telah melihat lalu lintas hanya dalam satu arah atau transaksi permintaan-respons tunggal.
  - Min: 30 detik
  - Maks: 60 detik
  - Default: 30 detik
- Batas waktu aliran UDP: Batas waktu (dalam detik) untuk alur UDP idle yang diklasifikasikan sebagai alur yang telah melihat lebih dari satu transaksi permintaan-respons.
  - Min: 60 detik
  - Maks: 180 detik
  - Default: 180 detik

13. Untuk Grup keamanan, pilih satu atau beberapa grup keamanan.
14. (Opsional) Untuk setiap tanda, pilih Tambahkan tanda baru dan masukkan kunci tanda dan nilai tanda opsional tersebut.
15. Pilih Buat antarmuka jaringan.

AWS CLI

Contoh 1: Untuk membuat antarmuka jaringan dengan alamat IP yang dipilih oleh Amazon EC2

Gunakan perintah create-network-interface berikut. Contoh ini membuat antarmuka jaringan dengan alamat IPv4 publik dan alamat IPv6 yang dipilih oleh Amazon EC2.

```
aws ec2 create-network-interface \
    --subnet-id subnet-0abcdef1234567890 \
    --description "my dual-stack network interface" \
    --ipv6-address-count 1 \
    --groups sg-1234567890abcdef0
```

Contoh 2: Untuk membuat antarmuka jaringan dengan alamat IP tertentu

Gunakan perintah create-network-interface berikut ini.

```
aws ec2 create-network-interface \
    --subnet-id subnet-0abcdef1234567890 \
    --description "my dual-stack network interface" \
    --private-ip-address 10.251.50.12 \
    --ipv6-addresses 2001:db8::1234:5678:1.2.3.4 \
    --groups sg-1234567890abcdef0
```

Contoh 3: Untuk membuat antarmuka jaringan dengan hitungan alamat IP sekunder

Gunakan perintah create-network-interface berikut ini. Dalam contoh ini, Amazon EC2 memilih alamat IP primer dan alamat IP sekunder.

```
aws ec2 create-network-interface \
    --subnet-id subnet-0abcdef1234567890 \
    --description "my network interface" \
    --secondary-private-ip-address-count 2 \
    --groups sg-1234567890abcdef0
```

Contoh 4: Untuk membuat antarmuka jaringan dengan alamat IP sekunder tertentu

Gunakan perintah create-network-interface berikut ini. Contoh ini menentukan alamat IP primer dan alamat IP sekunder.

```
aws ec2 create-network-interface \
    --subnet-id subnet-0abcdef1234567890 \
    --description "my network interface" \
    --private-ip-addresses PrivateIpAddress=10.0.1.30,Primary=true \
                           PrivateIpAddress=10.0.1.31,Primary=false
    --groups sg-1234567890abcdef0
```

PowerShell

Contoh 1: Untuk membuat antarmuka jaringan dengan alamat IP yang dipilih oleh Amazon EC2

Gunakan New-EC2NetworkInterface cmdlet. Contoh ini membuat antarmuka jaringan dengan alamat IPv4 publik dan alamat IPv6 yang dipilih oleh Amazon EC2.

```
New-EC2NetworkInterface `
    -SubnetId subnet-0abcdef1234567890 `
```

```
    -Description "my dual-stack network interface" `
    -Ipv6AddresCount 1 `
    -Group sg-1234567890abcdef0
```

Contoh 2: Untuk membuat antarmuka jaringan dengan alamat IP tertentu

Gunakan New-EC2NetworkInterface cmdlet.

```
New-EC2NetworkInterface `
    -SubnetId subnet-0abcdef1234567890 `
    -Description "my dual-stack network interface" `
    -PrivateIpAddress 10.251.50.12 `
    -Ipv6Address $ipv6addr `
    -Group sg-1234567890abcdef0
```

Tentukan alamat IPv6 sebagai berikut.

```
$ipv6addr = New-Object Amazon.EC2.Model.InstanceIpv6Address
$ipv6addr1.Ipv6Address = "2001:db8::1234:5678:1.2.3.4"
```

Contoh 3: Untuk membuat antarmuka jaringan dengan hitungan alamat IP sekunder

Gunakan New-EC2NetworkInterface cmdlet. Dalam contoh ini, Amazon EC2 memilih alamat IP primer dan alamat IP sekunder.

```
New-EC2NetworkInterface `
    -SubnetId subnet-0abcdef1234567890 `
    -Description "my network interface" `
    -SecondaryPrivateIpAddressCount 2 `
    -Group sg-1234567890abcdef0
```

Contoh 4: Untuk membuat antarmuka jaringan dengan alamat IP sekunder tertentu

Gunakan New-EC2NetworkInterface cmdlet. Contoh ini menentukan alamat IP primer dan alamat IP sekunder.

```
New-EC2NetworkInterface `
    -SubnetId subnet-0abcdef1234567890 `
    -Description "my network interface" `
    -PrivateIpAddresses @($primary, $secondary) `
    -Group sg-1234567890abcdef0
```

Tentukan alamat sekunder sebagai berikut.

```
$primary = New-Object Amazon.EC2.Model.PrivateIpAddressSpecification
$primary.PrivateIpAddress = "10.0.1.30"
$primary.Primary = $true
$secondary = New-Object Amazon.EC2.Model.PrivateIpAddressSpecification
$secondary.PrivateIpAddress = "10.0.1.31"
$secondary.Primary = $false
```

## Lampiran antarmuka jaringan untuk instans EC2 Anda

Anda dapat membuat antarmuka jaringan untuk digunakan oleh instans EC2 sebagai antarmuka jaringan primer atau sekunder. Anda harus melampirkan antarmuka jaringan ke instance EC2 jika berada di Zona Ketersediaan yang sama dengan antarmuka jaringan. Jenis instans menentukan berapa banyak antarmuka jaringan yang dapat Anda lampirkan ke instance. Untuk informasi selengkapnya, lihat the section called “Alamat IP per antarmuka jaringan”.

Pertimbangan

- Anda dapat memasang antarmuka jaringan ke sebuah instans saat berjalan (lampirkan panas), saat dihentikan (lampirkan hangat), atau saat instans diluncurkan (lampirkan dingin).
- Anda dapat melepaskan antarmuka jaringan sekunder saat instans sedang berjalan atau dihentikan. Namun, Anda tidak dapat melepaskan antarmuka jaringan primer.
- Anda dapat melepaskan antarmuka jaringan sekunder dari satu instance dan melampirkannya ke instance lain.
- Saat meluncurkan instans menggunakan CLI, API, atau SDK, Anda dapat menentukan antarmuka jaringan primer dan antarmuka jaringan tambahan. Perhatikan bahwa Anda tidak dapat mengaktifkan penetapan otomatis alamat IPv4 publik jika Anda menambahkan antarmuka jaringan sekunder selama peluncuran.
- Meluncurkan instans Amazon Linux atau Windows Server dengan beberapa antarmuka jaringan secara otomatis mengonfigurasi antarmuka, alamat IPv4 privat, dan tabel rute pada sistem operasi instans.
- Sambungan hangat atau panas dari antarmuka jaringan tambahan mungkin mengharuskan Anda untuk membuka antarmuka kedua secara manual, mengonfigurasi alamat IPv4 privat, dan memodifikasi tabel rute yang sesuai. Instans yang menjalankan Amazon Linux atau Windows Server secara otomatis mengenali warm atau hot attach dan mengonfigurasi sendiri.

- Anda tidak dapat melampirkan antarmuka jaringan lain ke instance (misalnya, konfigurasi tim NIC) untuk menambah atau menggandakan bandwidth jaringan ke atau dari instance dual-homed.
- Jika Anda melampirkan beberapa antarmuka jaringan dari subnet yang sama ke instance, Anda mungkin mengalami masalah jaringan seperti perutean asimetris. Jika memungkinkan, tambahkan alamat IPv4 pribadi sekunder pada antarmuka jaringan utama sebagai gantinya.
- Untuk instans EC2 di IPv6-only subnet, jika Anda melampirkan antarmuka jaringan sekunder, nama host DNS pribadi dari antarmuka jaringan sekunder diselesaikan ke alamat IPv6 utama untuk antarmuka jaringan utama.
- [Instans Windows] — Jika Anda menambahkan beberapa antarmuka jaringan ke sebuah instance, Anda harus mengonfigurasi antarmuka jaringan untuk menggunakan perutean statis.

## Lampirkan antarmuka jaringan

Anda dapat memasang antarmuka jaringan ke instans apa pun di Zona Ketersediaan yang sama seperti antarmuka jaringan, baik menggunakan halaman Instans atau Antarmuka Jaringan dari konsol Amazon EC2. Atau, Anda dapat menentukan antarmuka jaringan yang ada saat Anda meluncurkan instans.

Jika alamat IPv4 publik pada instans Anda dilepaskan, ia tidak menerima yang baru jika ada lebih dari satu antarmuka jaringan yang terpasang pada instans. Untuk informasi lebih lanjut tentang perilaku alamat IPv4 publik, lihat Alamat IPv4 publik.

Console

Untuk melampirkan antarmuka jaringan menggunakan halaman Instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih kotak centang untuk instans.
4. Pilih Tindakan, Jaringan, Lampirkan antarmuka jaringan.
5. Pilih VPC. Antarmuka jaringan dapat berada di VPC yang sama dengan instans Anda atau di VPC lain yang Anda miliki, selama antarmuka jaringan berada di Zona Ketersediaan yang sama dengan instans. Ini memungkinkan Anda membuat instans multi-homed di seluruh VPC dengan konfigurasi jaringan dan keamanan yang berbeda.
6. Pilih antarmuka jaringan. Jika instans mendukung beberapa kartu jaringan, Anda dapat memilih kartu jaringan.

7. Pilih Lampirkan.

Untuk melampirkan antarmuka jaringan menggunakan halaman Antarmuka Jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih kotak centang untuk antarmuka jaringan.
4. Pilih Tindakan, Lampirkan.
5. Pilih instans. Jika instans mendukung beberapa kartu jaringan, Anda dapat memilih kartu jaringan.
6. Pilih Lampirkan.

AWS CLI

Untuk melampirkan antarmuka jaringan

Gunakan perintah attach-network-interface berikut.

```
aws ec2 attach-network-interface \
    --network-interface-id eni-1234567890abcdef0 \
    --instance-id i-1234567890abcdef0 \
    --device-index 1
```

PowerShell

Untuk melampirkan antarmuka jaringan

Gunakan Add-EC2NetworkInterface cmdlet.

```
Add-EC2NetworkInterface `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -InstanceId i-1234567890abcdef0 `
    -DeviceIndex 1
```

## Lepaskan antarmuka jaringan

Anda dapat melepaskan antarmuka jaringan sekunder yang terpasang ke instans EC2 kapan saja, menggunakan halaman Instans atau Antarmuka Jaringan dari konsol Amazon EC2.

Jika Anda mencoba melepaskan antarmuka jaringan yang dilampirkan ke sumber daya dari layanan lain, seperti penyeimbang beban Elastic Load Balancing, fungsi Lambda, a WorkSpace, atau gateway NAT, Anda mendapatkan kesalahan bahwa Anda tidak memiliki izin untuk mengakses sumber daya. Untuk menemukan layanan mana yang menciptakan sumber daya yang melekat pada antarmuka jaringan, periksa deskripsi antarmuka jaringan. Jika Anda menghapus sumber daya, maka antarmuka jaringannya akan dihapus.

Console

Untuk melepaskan antarmuka jaringan menggunakan halaman Instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih kotak centang untuk instans. Periksa Antarmuka jaringan bagian tab Jaringan untuk memverifikasi bahwa antarmuka jaringan dilampirkan ke sebuah instans sebagai antarmuka jaringan sekunder.
4. Pilih Tindakan, Jaringan, Lepaskan antarmuka jaringan.
5. Pilih antarmuka jaringan dan pilih Lepaskan.

Untuk melepaskan antarmuka jaringan menggunakan halaman Antarmuka Jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih kotak centang untuk antarmuka jaringan. Periksa Detail instans bagian tab Detail untuk memverifikasi bahwa antarmuka jaringan dilampirkan ke sebuah instans sebagai antarmuka jaringan sekunder.
4. Pilih Tindakan, Lepaskan.
5. Saat diminta konfirmasi, pilih Lepaskan.
6. Jika antarmuka jaringan gagal untuk melepaskan dari instans, pilih Lepaskan paksa, Aktifkan lalu coba lagi. Kami menyarankan lepaskan paksa hanya sebagai pilihan terakhir. Memaksakan pelepasan dapat mencegah Anda melampirkan antarmuka jaringan yang berbeda pada indeks yang sama hingga Anda memulai ulang instans. Ini juga dapat mencegah metadata instans agar tidak mencerminkan bahwa antarmuka jaringan telah dilepaskan hingga Anda memulai ulang instans.

AWS CLI

Untuk melepaskan antarmuka jaringan

Gunakan perintah attach -network-interface berikut.

```
aws ec2 detach-network-interface --attachment-id eni-attach-016c93267131892c9
```

PowerShell

Untuk melepaskan antarmuka jaringan

Gunakan Dismount-EC2NetworkInterface cmdlet.

```
Dismount-EC2NetworkInterface -AttachmentId eni-attach-016c93267131892c9
```

## Kelola alamat IP untuk antarmuka jaringan Anda

Anda dapat mengelola alamat IP berikut untuk antarmuka jaringan Anda:

- Alamat IP Elastis (satu per alamat IPv4 privat)
- Alamat IPv4
- Alamat IPv6

Console

Untuk mengelola alamat IP dari antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih kotak centang untuk antarmuka jaringan.
4. Untuk mengelola alamat IPv4 dan IPv6, lakukan hal berikut:
   a. Pilih Tindakan, Kelola Alamat IP.
   b. Bentangkan antarmuka jaringan.
   c. Untuk alamat IPv4, edit alamat IP sesuai kebutuhan. Untuk menetapkan alamat IPv4 tambahan, pilih Tet apkan alamat IP baru dan kemudian tentukan alamat IPv4 dari rentang subnet atau biarkan AWS pilih satu untuk Anda.

d. (Dual stack atau IPv6 saja) Untuk alamat IPv6, edit alamat IP sesuai kebutuhan. Untuk menetapkan alamat IPv6 tambahan, pilih Tet apkan alamat IP baru dan kemudian tentukan alamat IPv6 dari rentang subnet atau biarkan AWS pilih satu untuk Anda.

e. Untuk menetapkan atau membatalkan penetapan alamat IPv4 publik ke antarmuka jaringan, pilih IP Auto-assign publik. Anda dapat mengaktifkan atau menonaktifkan ini untuk antarmuka jaringan apa pun, tetapi itu hanya memengaruhi antarmuka jaringan utama.

f. (Tumpukan ganda atau IPv6-only) Untuk Tetapkan IP IPv6 primer, pilih Aktifkan untuk menetapkan alamat IPv6 utama. GUA pertama yang terkait dengan antarmuka jaringan dipilih sebagai alamat IPv6 utama. Setelah menetapkan alamat IPv6 utama, Anda tidak dapat mengubahnya. Alamat ini adalah alamat IPv6 utama sampai instance dihentikan atau antarmuka jaringan terlepas.

g. Pilih Simpan.

5. Untuk mengaitkan alamat IP Elastis, lakukan hal berikut:

   a. Pilih Tindakan, Alamat Asosiasi.

   b. Untuk alamat IP Elastis, pilih alamat IP Elastis.

   c. Untuk Alamat IPv4 privat, pilih alamat IPv4 privat untuk dikaitkan dengan alamat IP Elastis.

   d. (Opsional) Pilih Izinkan alamat IP Elastis untuk dialihkan jika antarmuka jaringan saat ini terkait dengan instans lain atau antarmuka jaringan.

   e. Pilih Kaitkan.

6. Untuk memisahkan alamat IP Elastis, lakukan hal berikut:

   a. Pilih Tindakan, Pisahkan Alamat.

   b. Untuk Alamat IP publik, pilih alamat IP Elastis.

   c. Pilih Pisahkan.

AWS CLI

Untuk mengelola alamat IPv4

Gunakan perintah attach -private-ip-address berikut untuk menetapkan alamat IPv4.

```
aws ec2 assign-private-ip-addresses \
    --network-interface-id eni-1234567890abcdef0 \
```

```
    --private-ip-addresses 10.0.0.82
```

Gunakan perintah unattach-private-ip-address berikut untuk membatalkan penetapan alamat IPv4.

```
aws ec2 unassign-private-ip-addresses \
    --network-interface-id eni-1234567890abcdef0 \
    --private-ip-addresses 10.0.0.82
```

Untuk mengelola alamat IPv6

Gunakan perintah attach -ipv6-address berikut untuk menetapkan alamat IPv6.

```
aws ec2 assign-ipv6-addresses \
    --network-interface-id eni-1234567890abcdef0 \
    --ipv6-addresses 2001:db8:1234:1a00:9691:9503:25ad:1761
```

Gunakan perintah unattach-ipv6-address berikut untuk membatalkan penetapan alamat IPv6.

```
aws ec2 unassign-ipv6-addresses \
    --network-interface-id eni-1234567890abcdef0 \
    --ipv6-addresses 2001:db8:1234:1a00:9691:9503:25ad:1761
```

Untuk mengelola alamat IP Elastic untuk alamat IPv4 pribadi utama

Gunakan perintah asosiate-address berikut untuk mengaitkan alamat IP Elastic dengan alamat IPv4 pribadi utama.

```
aws ec2 associate-address \
    --allocation-id eipalloc-0b263919b6EXAMPLE \
    --network-interface-id eni-1234567890abcdef0
```

Gunakan perintah disasosiate-address berikut untuk memisahkan alamat IP Elastic dari alamat IPv4 pribadi utama.

```
aws ec2 disassociate-address --association-id eipassoc-2bebb745a1EXAMPLE
```

PowerShell

Untuk mengelola alamat IPv4

Gunakan Register-EC2PrivateIpAddress cmdlet untuk menetapkan alamat IPv4.

```
Register-EC2PrivateIpAddress `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -PrivateIpAddress 10.0.0.82
```

Gunakan Unregister-EC2PrivateIpAddress cmdlet untuk membatalkan penetapan alamat IPv4.

```
Unregister-EC2PrivateIpAddress `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -PrivateIpAddress 10.0.0.82
```

Untuk mengelola alamat IPv6

Gunakan Register-EC2Ipv6AddressList cmdlet untuk menetapkan alamat IPv6.

```
Register-EC2Ipv6AddressList `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Ipv6Address 2001:db8:1234:1a00:9691:9503:25ad:1761
```

Gunakan Unregister-EC2Ipv6AddressList cmdlet untuk membatalkan penetapan alamat IPv6.

```
Unregister-EC2Ipv6AddressList `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Ipv6Address 2001:db8:1234:1a00:9691:9503:25ad:1761
```

Untuk mengelola alamat IP Elastic untuk alamat IPv4 pribadi utama

Gunakan Register-EC2Address cmdlet untuk mengaitkan alamat IP Elastic dengan alamat IPv4 pribadi utama.

```
Register-EC2Address `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -AllocationId eipalloc-0b263919b6EXAMPLE
```

Gunakan Unregister-EC2Address cmdlet untuk memisahkan alamat IP Elastic dari alamat IPv4 pribadi utama.

```
Unregister-EC2Address -AssociationId eipassoc-2bebb745a1EXAMPLE
```

# Memodifikasi atribut antarmuka jaringan

Anda dapat mengubah atribut antarmuka jaringan berikut:

- Deskripsi
- Grup keamanan
- Hapus saat penghentian
- Source/destination periksa
- Waktu habis pelacakan koneksi idle

Pertimbangan-pertimbangan

Anda tidak dapat mengubah atribut antarmuka jaringan yang dikelola oleh pemohon.

Console

Untuk memodifikasi atribut antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih kotak centang untuk antarmuka jaringan.
4. Untuk mengubah deskripsi, lakukan hal berikut
   a. Pilih Tindakan, Ubah deskripsi.
   b. Untuk Deskripsi, masukkan deskripsi.
   c. Pilih Simpan.
5. Untuk mengubah grup keamanan, lakukan hal berikut:
   a. Pilih Tindakan, Ubah grup keamanan.
   b. Untuk grup keamanan terkait, tambahkan dan hapus grup keamanan sesuai kebutuhan. Grup keamanan dan antarmuka jaringan harus dibuat untuk VPC yang sama.
   c. Pilih Simpan.
6. Untuk mengubah perilaku penghentian, lakukan hal berikut:
   a. Pilih Tindakan, Ubah perilaku penghentian.
   b. Pilih atau hapus hapus pada penghentian, Aktifkan.

c. Pilih Simpan.

7. Untuk mengubah peng source/destination ecekan, lakukan hal berikut:

   a. Pilih T ind akan, Ubah source/dest cek.

   b. Pilih atau hapus Source/destination centang, Aktifkan.

   c. Pilih Simpan.

8. Untuk mengubah batas waktu pelacakan koneksi idle, lakukan hal berikut:

   a. Pilih T indakan, Ubah batas waktu pelacakan koneksi idle.

   b. Ubah nilai batas waktu sesuai kebutuhan. Untuk informasi selengkapnya, lihat Waktu habis pelacakan koneksi idle.

      - TCP menetapkan batas waktu: Batas waktu (dalam detik) untuk koneksi TCP idle dalam keadaan mapan.
        - Min: `60` detik
        - Maks: `432000` detik
        - Default: `350` detik untuk jenis instance Nitro v6 (tidak termasuk P6e-GB200); `432000` detik untuk semua jenis instance lainnya (termasuk P6e-GB200).
        - Direkomendasikan: Kurang dari `432000` detik
      - Batas waktu UDP: Batas waktu (dalam detik) untuk alur UDP idle yang telah melihat lalu lintas hanya dalam satu arah atau transaksi permintaan-respons tunggal.
        - Min: `30` detik
        - Maks: `60` detik
        - Default: `30` detik
      - Batas waktu aliran UDP: Batas waktu (dalam detik) untuk alur UDP idle yang diklasifikasikan sebagai alur yang telah melihat lebih dari satu transaksi permintaan-respons.
        - Min: `60` detik
        - Maks: `180` detik
        - Default: `180` detik

   c. Pilih Simpan.

AWS CLI

Example Contoh: Untuk memodifikasi deskripsi

Gunakan perintah modify- network-interface-atribut berikut.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
    --description "my updated description"
```

Example Contoh: Untuk memodifikasi grup keamanan

Gunakan perintah modify- network-interface-atribut berikut.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
    --groups sg-1234567890abcdef0
```

Example Contoh: Untuk mengubah perilaku penghentian

Gunakan perintah modify- network-interface-atribut berikut.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
    --attachment AttachmentId=eni-attach-43348162abEXAMPLE,DeleteOnTermination=false
```

Example Contoh: Untuk mengaktifkan peng source/destination ecekan

Gunakan perintah modify- network-interface-atribut berikut.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
    --source-dest-check
```

Example Contoh: Untuk memodifikasi timout pelacakan koneksi idle

Gunakan perintah modify- network-interface-atribut berikut. Untuk informasi selengkapnya, lihat Waktu habis pelacakan koneksi idle.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
```

```
    --connection-tracking-specification
 TcpEstablishedTimeout=172800,UdpStreamTimeout=90,UdpTimeout=60
```

PowerShell

Example Contoh: Untuk memodifikasi deskripsi

Gunakan Edit-EC2NetworkInterfaceAttribute cmdlet.

```
Edit-EC2NetworkInterfaceAttribute `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Description "my updated description"
```

Example Contoh: Untuk memodifikasi grup keamanan

Gunakan Edit-EC2NetworkInterfaceAttribute cmdlet.

```
Edit-EC2NetworkInterfaceAttribute `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Group sg-1234567890abcdef0
```

Example Contoh: Untuk mengubah perilaku penghentian

Gunakan Edit-EC2NetworkInterfaceAttribute cmdlet.

```
Edit-EC2NetworkInterfaceAttribute `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Attachment_AttachmentId eni-attach-43348162abEXAMPLE `
    -Attachment_DeleteOnTermination $false
```

Example Contoh: Untuk mengaktifkan peng source/destination ecekan

Gunakan Edit-EC2NetworkInterfaceAttribute cmdlet.

```
Edit-EC2NetworkInterfaceAttribute `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -SourceDestCheck $true
```

Example Contoh: Untuk mengubah batas waktu pelacakan koneksi idle

Gunakan Edit-EC2NetworkInterfaceAttribute cmdlet. Untuk informasi selengkapnya, lihat Waktu habis pelacakan koneksi idle.

```
Edit-EC2NetworkInterfaceAttribute `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -ConnectionTrackingSpecification_TcpEstablishedTimeout 172800 `
    -ConnectionTrackingSpecification_UdpStreamTimeout 90 `
    -ConnectionTrackingSpecification_UdpTimeout 60
```

## Beberapa antarmuka jaringan untuk instans Amazon EC2 Anda

Melampirkan beberapa antarmuka jaringan ke instance berguna ketika Anda memerlukan yang berikut:

- Jaringan manajemen.
- Jaringan dan peralatan keamanan.
- Dual-homed instance dengan beban kerja di subnet atau VPC yang berbeda.
- Solusi dengan anggaran rendah dan ketersediaan tinggi.

### Jaringan manajemen

Ikhtisar berikut menjelaskan jaringan manajemen yang dibuat menggunakan beberapa antarmuka jaringan.

Kriteria

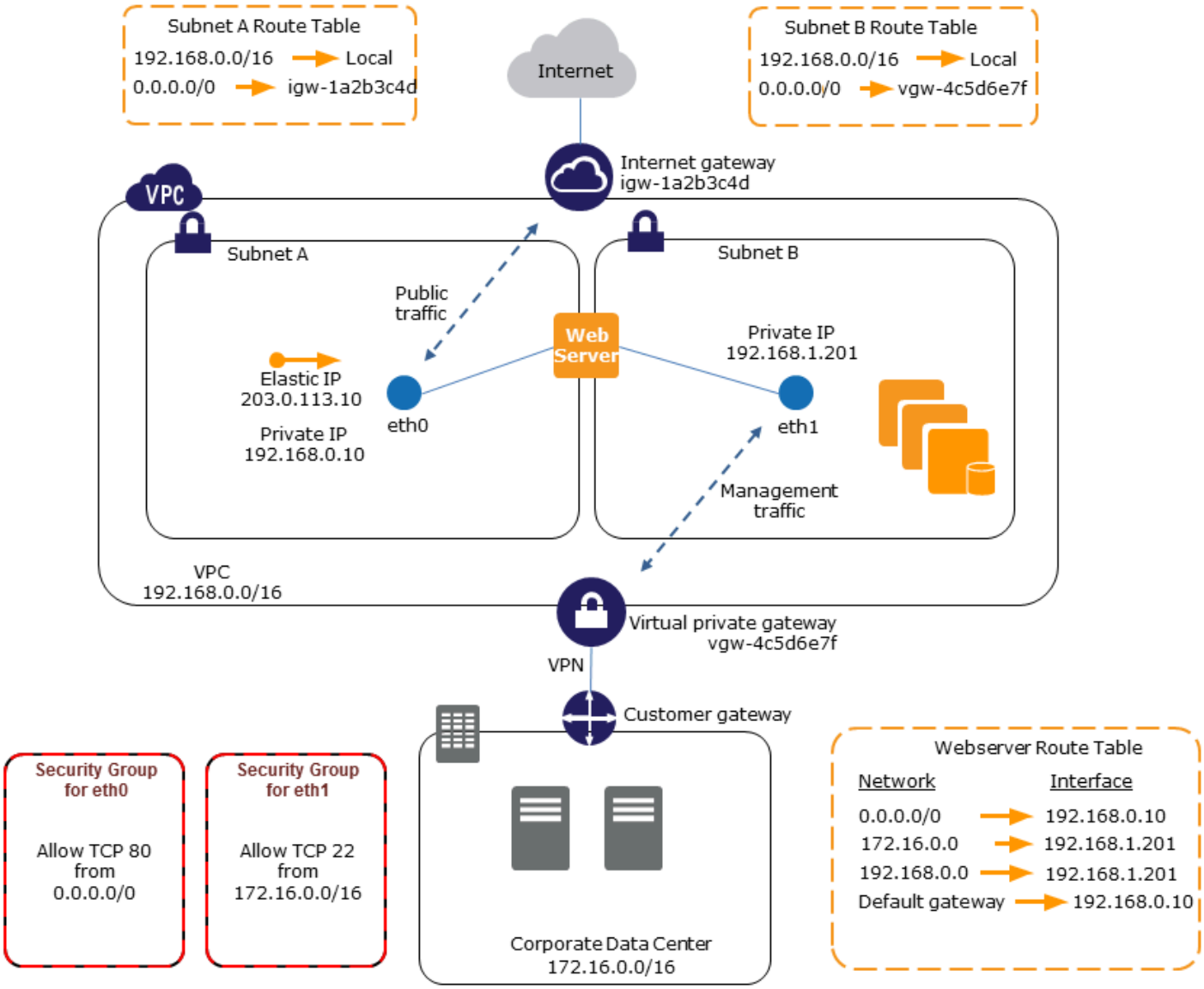
- Antarmuka jaringan utama pada instance (misalnya,eth0) menangani lalu lintas publik.
- Antarmuka jaringan sekunder pada instance (misalnya,eth1) menangani lalu lintas manajemen backend. Ini terhubung ke subnet terpisah yang memiliki kontrol akses yang lebih ketat, dan terletak di Zona Ketersediaan yang sama dengan antarmuka jaringan utama.

Pengaturan

- Antarmuka jaringan primer, yang mungkin atau mungkin tidak berada di belakang penyeimbang beban, memiliki grup keamanan terkait yang memungkinkan akses ke server dari internet. Misalnya, izinkan port TCP 80 dan 443 dari 0.0.0. 0/0 atau dari penyeimbang beban.
- Antarmuka jaringan sekunder memiliki grup keamanan terkait yang memungkinkan akses SSH saja, dimulai dari salah satu lokasi berikut:
  - Rentang alamat IP yang diizinkan, baik di dalam VPC, atau dari internet.

- Subnet pribadi dalam Availability Zone yang sama dengan antarmuka jaringan utama.
- Virtual private gateway.

> **Note**
>
> Untuk memastikan kemampuan failover, pertimbangkan untuk menggunakan IPv4 privat sekunder untuk lalu lintas masuk pada antarmuka jaringan. Jika terjadi kegagalan instance, Anda dapat memindahkan alamat IPv4 pribadi and/or sekunder antarmuka ke instance siaga.



## Peralatan jaringan dan keamanan

Beberapa peralatan jaringan dan keamanan, seperti penyeimbang beban, server network address translation (NAT), dan server proksi lebih suka dikonfigurasi dengan beberapa antarmuka jaringan. Anda dapat membuat dan melampirkan antarmuka jaringan sekunder ke instans yang menjalankan jenis aplikasi ini dan mengonfigurasi antarmuka tambahan dengan alamat IP publik dan pribadi, grup keamanan, dan source/destination pemeriksaan mereka sendiri.

## Dual-homed instance dengan beban kerja di subnet yang berbeda

Anda dapat menempatkan antarmuka jaringan di setiap server web Anda yang terhubung ke jaringan tingkat menengah tempat server aplikasi berada. Server aplikasi juga bisa menjadi dual-homed ke jaringan backend (subnet) tempat server basis data berada. Alih-alih merutekan paket jaringan melalui instans dual-homed, setiap instans dual-homed menerima dan memproses permintaan di front end, memulai koneksi ke backend, dan kemudian mengirim permintaan ke server di jaringan backend.

## Dual-homed instance dengan beban kerja di VPC yang berbeda di akun yang sama

Anda dapat meluncurkan instans EC2 dalam satu VPC dan melampirkan ENI sekunder dari VPC yang berbeda, selama antarmuka jaringan berada di Zona Ketersediaan yang sama dengan instans. Ini memungkinkan Anda membuat instans multi-homed di seluruh VPC dengan konfigurasi jaringan dan keamanan yang berbeda. Anda tidak dapat membuat instans multi-homed di seluruh VPC di akun yang berbeda. AWS

Anda dapat menggunakan instans dual-homed di seluruh VPC dalam kasus penggunaan berikut ini:

- Atasi tumpang tindih CIDR antara dua VPC yang tidak dapat diintip bersama: Anda dapat memanfaatkan CIDR sekunder dalam VPC dan mengizinkan instans untuk berkomunikasi di dua rentang IP yang tidak tumpang tindih.
- Hubungkan beberapa VPC dalam satu akun: Aktifkan komunikasi antara sumber daya individu yang biasanya dipisahkan oleh batas VPC.

## Low-budget, solusi ketersediaan tinggi

Jika salah satu instans Anda yang melayani fungsi tertentu gagal, antarmuka jaringannya dapat dilampirkan ke instans pengganti atau hot standby yang telah dikonfigurasi sebelumnya untuk peran yang sama untuk memulihkan layanan dengan cepat. Misalnya, Anda dapat menggunakan

antarmuka jaringan sebagai antarmuka jaringan primer atau sekunder ke layanan penting seperti instans basis data atau instans NAT. Jika instans gagal, Anda (atau lebih mungkin, kode yang berjalan atas nama Anda) dapat memasang antarmuka jaringan ke instans hot standby. Karena antarmuka mempertahankan alamat IP privatnya, alamat IP Elastis, dan alamat MAC, lalu lintas jaringan mulai mengalir ke instans siaga segera setelah Anda memasang antarmuka jaringan ke instans pengganti. Pengguna mengalami kehilangan konektivitas singkat antara waktu instans gagal dan waktu saat antarmuka jaringan dilampirkan ke instans standby, tetapi tidak ada perubahan pada tabel rute VPC atau server DNS Anda yang diperlukan.

## Requester-managed antarmuka jaringan

Antarmuka jaringan yang dikelola pemohon adalah antarmuka jaringan yang dibuat di VPC Layanan AWS Anda atas nama Anda. Antarmuka jaringan dikaitkan dengan sumber daya untuk layanan lain, seperti instans DB dari Amazon RDS, gateway NAT, atau titik akhir VPC antarmuka dari. AWS PrivateLink

Pertimbangan-pertimbangan

- Anda dapat melihat antarmuka jaringan yang dikelola pemohon di akun Anda dan Anda dapat menambahkan atau menghapus tag. Namun, Anda tidak dapat melakukan tindakan berikut pada antarmuka jaringan yang dikelola oleh pemohon:
  - Memodifikasi atribut antarmuka jaringan
  - Pasang atau lepaskan antarmuka jaringan
  - Mengaitkan atau memisahkan alamat IP Elastic
  - Menetapkan atau membatalkan penetapan alamat IP pribadi
  - Tentukan antarmuka jaringan saat meluncurkan instance

  Anda masih dapat mengatur ulang atribut antarmuka jaringan, meskipun Anda tidak dapat memodifikasinya.
- Saat Anda menghapus sumber daya yang terkait dengan antarmuka jaringan yang dikelola pemohon, antarmuka jaringan akan Layanan AWS melepaskan dan menghapusnya. Jika layanan melepaskan antarmuka jaringan tetapi tidak menghapusnya, Anda dapat menghapus antarmuka jaringan yang terpisah.

### Console

Untuk melihat antarmuka jaringan yang dikelola oleh pemohon

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Jaringan & Keamanan, Antarmuka Jaringan.
3. Pilih ID antarmuka jaringan untuk membuka halaman detailnya.
4. Berikut ini adalah bidang kunci yang dapat Anda gunakan untuk menentukan tujuan antarmuka jaringan:

   - Deskripsi: Deskripsi yang disediakan oleh AWS layanan yang menciptakan antarmuka. Misalnya, “VPC Endpoint Interface vpce 089f2123488812123”.
   - Requester-managed: Menunjukkan apakah antarmuka jaringan dikelola oleh AWS.
   - ID Pemohon: Alias atau ID AWS akun dari prinsipal atau layanan yang membuat antarmuka jaringan. Jika Anda membuat antarmuka jaringan, ini adalah Akun AWS ID Anda. Jika tidak, pengguna utama atau layanan lain menciptakannya.

### AWS CLI

Untuk melihat antarmuka jaringan yang dikelola oleh pemohon

Gunakan perintah describe-network-interfaces sebagai berikut.

```
aws ec2 describe-network-interfaces \
    --filters "Name=requester-managed,Values=true" \
    --query "NetworkInterfaces[*].[Description, InterfaceType]" \
    --output table
```

Berikut ini adalah contoh output yang menunjukkan bidang kunci yang dapat Anda gunakan untuk menentukan tujuan antarmuka jaringan: `Description` dan `InterfaceType`.

```
-------------------------------------------------------------------------------
|                          DescribeNetworkInterfaces                          |
+-----------------------------------------------------+-------------------------+
|  VPC Endpoint Interface: vpce-0f00567fa8477a1e6     |  interface              |
|  VPC Endpoint Interface vpce-0d8ddce4be80e4474      |  interface              |
|  VPC Endpoint Interface vpce-078221a1e27d1ea5b      |  vpc_endpoint           |
|  Resource Gateway Interface rgw-0bba03f3d56060135   |  interface              |
|  VPC Endpoint Interface: vpce-0cc199f605eaeace7     |  interface              |
```

```
|  VPC Endpoint Interface vpce-019b90d6f16d4f958    |  interface              |
+---------------------------------------------------+-------------------------+
```

PowerShell

Untuk melihat antarmuka jaringan yang dikelola oleh pemohon

Gunakan Get-EC2NetworkInterface cmdlet sebagai berikut.

```
Get-EC2NetworkInterface -Filter @{Name="requester-managed"; Values="true"} | Select
 Description, InterfaceType
```

Berikut ini adalah contoh output yang menunjukkan bidang kunci yang dapat Anda gunakan untuk menentukan tujuan antarmuka jaringan: `Description` dan`InterfaceType`.

```
Description                                       InterfaceType
-----------                                       -------------
VPC Endpoint Interface: vpce-0f00567fa8477a1e6    interface
VPC Endpoint Interface vpce-0d8ddce4be80e4474     interface
VPC Endpoint Interface vpce-078221a1e27d1ea5b     vpc_endpoint
Resource Gateway Interface rgw-0bba03f3d56060135  interface
VPC Endpoint Interface: vpce-0cc199f605eaeace7    interface
VPC Endpoint Interface vpce-019b90d6f16d4f958     interface
```

## Delegasi awalan untuk antarmuka jaringan Amazon EC2

Anda dapat menetapkan rentang CIDR IPv4 atau IPv6 privat, baik secara otomatis atau manual, ke antarmuka jaringan. Dengan menetapkan prefiks, Anda menskalakan dan menyederhanakan manajemen aplikasi, termasuk aplikasi kontainer dan jaringan yang memerlukan beberapa alamat IP pada sebuah instans. Untuk informasi selengkapnya tentang alamat IPv4 dan IPv6, lihat Pengalamatan IP instans Amazon EC2.

Pilihan penetapan berikut tersedia:

- Penugasan otomatis — AWS memilih awalan dan menetapkannya ke antarmuka jaringan Anda. Jika subnet untuk antarmuka jaringan memiliki reservasi tipe CIDR subnet`prefix`, kami memilih awalan dari reservasi subnet CIDR. Jika tidak, kami memilihnya dari kisaran subnet CIDR.
- Penugasan manual — Anda menentukan awalan dan AWS memverifikasi bahwa itu belum ditetapkan ke sumber daya lain sebelum menetapkannya ke antarmuka jaringan Anda.

Menetapkan prefiks memiliki manfaat sebagai berikut:

- Peningkatan alamat IP pada antarmuka jaringan — Ketika Anda menggunakan prefiks, Anda menetapkan blok alamat IP sebagai lawan dari alamat IP individual. Ini meningkatkan jumlah alamat IP untuk antarmuka jaringan.
- Manajemen VPC yang disederhanakan untuk kontainer — Dalam aplikasi kontainer, setiap kontainer memerlukan alamat IP yang unik. Menetapkan prefiks ke instans akan menyederhanakan manajemen VPC, karena Anda dapat meluncurkan dan menghentikan kontainer tanpa harus memanggil Amazon EC2 API untuk penetapan IP individual.

Daftar Isi


- Hal-hal mendasar
- Pertimbangan-pertimbangan
- Kelola awalan untuk antarmuka jaringan Anda
  - Tetapkan prefiks selama pembuatan antarmuka jaringan
  - Tetapkan awalan ke antarmuka jaringan yang ada
  - Hapus prefiks dari antarmuka jaringan Anda


## Hal-hal mendasar

- Anda dapat menetapkan prefiks ke antarmuka jaringan baru atau yang sudah ada.
- Untuk menggunakan prefiks, Anda menetapkan prefiks ke antarmuka jaringan Anda, melampirkan antarmuka jaringan ke instans Anda, lalu mengonfigurasi sistem operasi Anda.
- Saat Anda memilih opsi untuk menentukan prefiks, prefiks harus memenuhi persyaratan berikut ini:
  - Prefiks IPv4 yang dapat Anda tentukan adalah. `/28`
  - Prefiks IPv6 yang dapat Anda tentukan adalah. `/80`
  - Prefiks ada di subnet CIDR antarmuka jaringan, dan tidak tumpang tindih dengan prefiks lain atau alamat IP yang ditetapkan ke sumber daya yang ada di subnet.
- Anda dapat menetapkan prefiks ke antarmuka jaringan primer atau sekunder.
- Anda dapat menetapkan alamat IP Elastis ke antarmuka jaringan yang memiliki prefiks yang ditetapkan untuk itu.
- Anda juga dapat menetapkan alamat IP Elastis ke bagian alamat IP dari prefiks yang ditetapkan.
- Kami menyelesaikan nama host DNS privat sebuah instans ke alamat IPv4 privat primer.

- Kami menetapkan setiap alamat IPv4 privat untuk antarmuka jaringan, termasuk yang dari prefiks, menggunakan format berikut:
  - Wilayah `us-east-1`

    ```
    ip-private-ipv4-address.ec2.internal
    ```

  - Wilayah Lainnya

    ```
    ip-private-ipv4-address.region.compute.internal
    ```

## Pertimbangan-pertimbangan

Pertimbangkan hal berikut ini saat Anda menggunakan prefiks:

- Antarmuka jaringan dengan awalan didukung dengan Nitro-based instance.
- Awalan untuk antarmuka jaringan harus menggunakan alamat IPv6 atau alamat IPv4 pribadi.
- Jumlah maksimum alamat IP yang dapat Anda tetapkan ke antarmuka jaringan tergantung pada tipe instans. Setiap prefiks yang Anda tetapkan ke antarmuka jaringan dihitung sebagai satu alamat IP. Misalnya, `c5.large` instans memiliki batas alamat `10` IPv4 per antarmuka jaringan. Setiap antarmuka jaringan untuk instans ini memiliki alamat IPv4 primer. Jika antarmuka jaringan tidak memiliki alamat IPv4 sekunder, Anda dapat menetapkan hingga 9 prefiks ke antarmuka jaringan. Untuk setiap alamat IPv4 tambahan yang Anda tetapkan ke antarmuka jaringan, Anda dapat menetapkan satu prefiks kurang ke antarmuka jaringan. Untuk informasi selengkapnya, lihat Alamat IP maksimum per antarmuka jaringan.
- Awalan disertakan dalam source/destination cek.
- Anda harus mengonfigurasi sistem operasi Anda untuk bekerja dengan antarmuka jaringan dengan prefiks. Perhatikan hal-hal berikut:
  - Beberapa AMI Amazon Linux berisi skrip tambahan yang diinstal oleh AWS, yang dikenal sebagai`ec2-net-utils`. Skrip ini secara opsional mengotomatiskan konfigurasi antarmuka jaringan Anda. Mereka hanya untuk digunakan di Amazon Linux.
  - Untuk kontainer, Anda dapat menggunakan Container Network Interface (CNI) untuk plug-in Kubernetes, atau `dockerd` jika Anda menggunakan Docker untuk mengelola kontainer Anda.

## Kelola awalan untuk antarmuka jaringan Anda

Saat Anda menetapkan awalan ke antarmuka jaringan, Anda dapat memilih apakah mengizinkan kami menetapkan awalan secara otomatis atau Anda dapat menentukan awalan khusus. Jika Anda membiarkan kami secara otomatis menetapkan awalan dan subnet untuk antarmuka jaringan memiliki reservasi tipe CIDR subnet`prefix`, kami memilih awalan dari reservasi CIDR subnet. Jika tidak, kami memilihnya dari kisaran subnet CIDR.

Tugas

- Tetapkan prefiks selama pembuatan antarmuka jaringan
- Tetapkan awalan ke antarmuka jaringan yang ada
- Hapus prefiks dari antarmuka jaringan Anda

### Tetapkan prefiks selama pembuatan antarmuka jaringan

Anda dapat menetapkan awalan otomatis atau kustom saat membuat antarmuka jaringan.

Console

Untuk menetapkan prefiks otomatis selama pembuatan antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih Buat antarmuka jaringan.
4. Masukkan deskripsi untuk antarmuka jaringan, pilih subnet tempat untuk membuat antarmuka jaringan, dan konfigurasikan alamat IPv4 dan IPv6 pribadi.
5. Perluas Pengaturan lanjutan.
6. Untuk delegasi awalan IPv4 lakukan salah satu hal berikut:
   - Untuk menetapkan awalan IPv4 secara otomatis, pilih Auto-assign. Untuk Jumlah prefiks IPv4, masukkan jumlah awalan yang akan ditetapkan.
   - Untuk menetapkan awalan IPv4 tertentu, pilih K ustom. Pilih Tambahkan awalan baru dan masukkan awalan.
7. Untuk delegasi awalan IPv6 lakukan salah satu hal berikut:
   - Untuk menetapkan awalan IPv6 secara otomatis, pilih Auto-assign. Untuk Jumlah prefiks IPv6, masukkan jumlah awalan yang akan ditetapkan.

- Untuk menetapkan awalan IPv6 tertentu, pilih K ustom. Pilih Tambahkan awalan baru dan masukkan awalan.

> **Note**
>
> Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

8. Pilih grup keamanan yang akan dikaitkan dengan antarmuka jaringan dan tetapkan tanda sumber daya jika diperlukan.
9. Pilih Buat antarmuka jaringan.

AWS CLI

Untuk menetapkan prefiks IPv4 otomatis selama pembuatan antarmuka jaringan

Gunakan perintah create-network-interface dan atur `--ipv4-prefix-count` ke jumlah prefiks IPv4 untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv4.

```
aws ec2 create-network-interface \
    --subnet-id subnet-047cfed18eEXAMPLE \
    --description "IPv4 automatic example" \
    --ipv4-prefix-count 1
```

Untuk menetapkan prefiks IPv4 tertentu selama pembuatan antarmuka jaringan

Gunakan perintah create-network-interface dan atur `--ipv4-prefixes` ke awalan. AWS memilih alamat IPv4 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 10.0.0.208/28.

```
aws ec2 create-network-interface \
    --subnet-id subnet-047cfed18eEXAMPLE \
    --description "IPv4 manual example" \
    --ipv4-prefixes Ipv4Prefix=10.0.0.208/28
```

Untuk menetapkan prefiks IPv6 otomatis selama pembuatan antarmuka jaringan

Gunakan perintah create-network-interface dan atur `--ipv6-prefix-count` ke jumlah awalan IPv6 untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv6.

```
aws ec2 create-network-interface \
```

```
    --subnet-id subnet-047cfed18eEXAMPLE \
    --description "IPv6 automatic example" \
    --ipv6-prefix-count 1
```

Untuk menetapkan prefiks IPv6 tertentu selama pembuatan antarmuka jaringan

Gunakan perintah create-network-interface dan atur `--ipv6-prefixes` ke awalan. AWS memilih alamat IPv6 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 2600:1f13:fc2:a700:1768::/80.

```
aws ec2 create-network-interface \
    --subnet-id subnet-047cfed18eEXAMPLE \
    --description "IPv6 manual example" \
    --ipv6-prefixes Ipv6Prefix=2600:1f13:fc2:a700:1768::/80
```

PowerShell

Untuk menetapkan prefiks IPv4 otomatis selama pembuatan antarmuka jaringan

Gunakan New-EC2NetworkInterface cmdlet dan atur `Ipv4PrefixCount` ke jumlah prefiks IPv4 untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv4.

```
New-EC2NetworkInterface `
    -SubnetId 'subnet-047cfed18eEXAMPLE' `
    -Description 'IPv4 automatic example' `
    -Ipv4PrefixCount 1
```

Untuk menetapkan prefiks IPv4 tertentu selama pembuatan antarmuka jaringan

Gunakan New-EC2NetworkInterface cmdlet dan atur `Ipv4Prefix` ke awalan. AWS memilih alamat IPv4 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 10.0.0.208/28.

```
Import-Module AWS.Tools.EC2
New-EC2NetworkInterface `
    -SubnetId 'subnet-047cfed18eEXAMPLE' `
    -Description 'IPv4 manual example' `
    -Ipv4Prefix (New-Object `
        -TypeName Amazon.EC2.Model.Ipv4PrefixSpecificationRequest `
        -Property @{Ipv4Prefix = '10.0.0.208/28'})
```

Untuk menetapkan prefiks IPv6 otomatis selama pembuatan antarmuka jaringan

Gunakan New-EC2NetworkInterface cmdlet dan atur `Ipv6PrefixCount` ke jumlah awalan IPv6 untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv6.

```
New-EC2NetworkInterface `
    -SubnetId 'subnet-047cfed18eEXAMPLE' `
    -Description 'IPv6 automatic example' `
    -Ipv6PrefixCount 1
```

Untuk menetapkan prefiks IPv6 tertentu selama pembuatan antarmuka jaringan

Gunakan New-EC2NetworkInterface cmdlet dan atur `Ipv6Prefixes` ke awalan. AWS memilih alamat IPv6 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 2600:1f13:fc2:a700:1768::/80.

```
Import-Module AWS.Tools.EC2
New-EC2NetworkInterface `
    -SubnetId 'subnet-047cfed18eEXAMPLE' `
    -Description 'IPv6 manual example' `
    -Ipv6Prefix (New-Object `
        -TypeName Amazon.EC2.Model.Ipv6PrefixSpecificationRequest `
        -Property @{Ipv6Prefix = '2600:1f13:fc2:a700:1768::/80'})
```

Tetapkan awalan ke antarmuka jaringan yang ada

Anda dapat menetapkan awalan otomatis atau kustom ke antarmuka jaringan yang ada.

Console

Untuk menetapkan prefiks otomatis ke antarmuka jaringan yang ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih antarmuka jaringan yang akan menetapkan prefiks, dan pilih Tindakan, Kelola prefiks.
4. Untuk delegasi awalan IPv4 lakukan salah satu hal berikut:
   - Untuk menetapkan awalan IPv4 secara otomatis, pilih Auto-assign. Untuk Jumlah prefiks IPv4, masukkan jumlah awalan yang akan ditetapkan.
   - Untuk menetapkan awalan IPv4 tertentu, pilih K ustom. Pilih Tambahkan awalan baru dan masukkan awalan.

5. Untuk delegasi awalan IPv6 lakukan salah satu hal berikut:
   - Untuk menetapkan awalan IPv6 secara otomatis, pilih Auto-assign. Untuk Jumlah prefiks IPv6, masukkan jumlah awalan yang akan ditetapkan.
   - Untuk menetapkan awalan IPv6 tertentu, pilih K ustom. Pilih Tambahkan awalan baru dan masukkan awalan.

   > **Note**
   >
   > Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

6. Pilih Simpan.

AWS CLI

Gunakan perintah attach -ipv6-address untuk menetapkan awalan IPv6 dan perintah attach -private-ip-address untuk menetapkan awalan IPv4 ke antarmuka jaringan yang ada.

Untuk menetapkan prefiks IPv4 otomatis ke antarmuka jaringan yang ada

Gunakan perintah attach -private-ip-address dan atur ke jumlah prefiks IPv4 `--ipv4-prefix-count` untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv4.

```
aws ec2 assign-private-ip-addresses \
    --network-interface-id eni-081fbb4095EXAMPLE \
    --ipv4-prefix-count 1
```

Untuk menetapkan prefiks IPv4 tertentu ke antarmuka jaringan yang ada

Gunakan perintah assign-private-ip-addresses dan atur `--ipv4-prefixes` ke prefiks. AWS memilih alamat IPv4 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 10.0.0.208/28.

```
aws ec2 assign-private-ip-addresses \
    --network-interface-id eni-081fbb4095EXAMPLE \
    --ipv4-prefixes 10.0.0.208/28
```

Untuk menetapkan prefiks IPv6 otomatis ke antarmuka jaringan yang ada

Gunakan perintah attach -ipv6-address dan atur ke jumlah prefiks IPv6 `--ipv6-prefix-count` untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv6.

```
aws ec2 assign-ipv6-addresses \
    --network-interface-id eni-00d577338cEXAMPLE \
    --ipv6-prefix-count 1
```

Untuk menetapkan prefiks IPv6 tertentu ke antarmuka jaringan yang ada

Gunakan perintah Assign-ipv6-address dan atur ke awalan. `--ipv6-prefixes` AWS memilih alamat IPv6 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 2600:1f13:fc2:a700:18bb::/80.

```
aws ec2 assign-ipv6-addresses \
    --network-interface-id eni-00d577338cEXAMPLE \
    --ipv6-prefixes 2600:1f13:fc2:a700:18bb::/80
```

PowerShell

Untuk menetapkan prefiks IPv4 otomatis ke antarmuka jaringan yang ada

Gunakan Register-EC2PrivateIpAddress cmdlet dan atur `Ipv4PrefixCount` ke jumlah prefiks IPv4 untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv4.

```
Register-EC2PrivateIpAddress `
    -NetworkInterfaceId 'eni-00d577338cEXAMPLE' `
    -Ipv4PrefixCount 1
```

Untuk menetapkan prefiks IPv4 tertentu ke antarmuka jaringan yang ada

Gunakan Register-EC2PrivateIpAddress cmdlet dan atur `Ipv4Prefix` ke awalan. AWS memilih alamat IPv4 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 10.0.0.208/28.

```
Register-EC2PrivateIpAddress `
    -NetworkInterfaceId 'eni-00d577338cEXAMPLE' `
    -Ipv4Prefix '10.0.0.208/28'
```

Untuk menetapkan prefiks IPv6 otomatis ke antarmuka jaringan yang ada

Gunakan Register-EC2Ipv6AddressList cmdlet dan atur `Ipv6PrefixCount` ke jumlah prefiks IPv4 untuk ditetapkan. AWS Dalam contoh berikut, AWS menetapkan satu awalan IPv6.

```
Register-EC2Ipv6AddressList `
```

```
    -NetworkInterfaceId 'eni-00d577338cEXAMPLE' `
    -Ipv6PrefixCount 1
```

Untuk menetapkan prefiks IPv6 tertentu ke antarmuka jaringan yang ada

Gunakan Register-EC2Ipv6AddressList cmdlet dan atur `Ipv6Prefix` ke awalan. AWS memilih alamat IPv6 dari rentang ini. Dalam contoh berikut, prefiks CIDR adalah 2600:1f13:fc2:a700:18bb::/80.

```
Register-EC2Ipv6AddressList `
    -NetworkInterfaceId 'eni-00d577338cEXAMPLE' `
    -Ipv6Prefix '2600:1f13:fc2:a700:18bb::/80'
```

Hapus prefiks dari antarmuka jaringan Anda

Anda dapat menghapus awalan dari antarmuka jaringan yang ada.

Console

Untuk menghapus prefiks dari antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih antarmuka jaringan.
4. Pilih T indakan, Kelola awalan.
5. Untuk delegasi awalan IPv4, untuk menghapus awalan tertentu, pilih Batalkan penetapan di sebelah awalan yang akan dihapus. Untuk menghapus semua awalan, pilih Jangan tetapkan.
6. Untuk delegasi awalan IPv6, untuk menghapus awalan tertentu, pilih Batalkan penetapan di sebelah awalan yang akan dihapus. Untuk menghapus semua awalan, pilih Jangan tetapkan.

   > **Note**
   >
   > Delegasi prefiks IPv6 hanya muncul jika subnet yang dipilih diaktifkan untuk IPv6.

7. Pilih Simpan.

AWS CLI

Anda dapat menggunakan perintah unassign-ipv6-addresses untuk menghapus prefiks IPv6 dan perintah unassign-private-ip-addresses untuk menghapus prefiks IPv4 dari antarmuka jaringan yang ada.

Untuk menghapus prefiks IPv4 dari antarmuka jaringan

Gunakan perintah unassign-private-ip-address dan atur `--ipv4-prefix` ke awalan CIDR untuk dihapus.

```
aws ec2 unassign-private-ip-addresses \
    --network-interface-id eni-081fbb4095EXAMPLE \
    --ipv4-prefixes 10.0.0.176/28
```

Untuk menghapus prefiks IPv6 dari antarmuka jaringan

Gunakan perintah unassign-ipv6-address dan atur `--ipv6-prefix` ke awalan CIDR untuk dihapus.

```
aws ec2 unassign-ipv6-addresses \
    --network-interface-id eni-00d577338cEXAMPLE \
    --ipv6-prefix 2600:1f13:fc2:a700:18bb::/80
```

PowerShell

Untuk menghapus prefiks IPv4 dari antarmuka jaringan

Gunakan Unregister-EC2PrivateIpAddress cmdlet dan atur `Ipv4Prefix` ke awalan CIDR untuk menghapus.

```
Unregister-EC2PrivateIpAddress `
    -NetworkInterfaceId 'eni-00d577338cEXAMPLE' `
    -Ipv4Prefix '10.0.0.208/28'
```

Untuk menghapus prefiks IPv6 dari antarmuka jaringan

Gunakan Unregister-EC2Ipv6AddressList cmdlet dan atur `Ipv6Prefix` ke awalan CIDR untuk menghapus.

```
Unregister-EC2Ipv6AddressList `
    -NetworkInterfaceId 'eni-00d577338cEXAMPLE' `
    -Ipv6Prefix '2600:1f13:fc2:a700:18bb::/80'
```

## Menghapus antarmuka jaringan

Menghapus antarmuka jaringan melepaskan semua atribut yang terkait dengan antarmuka dan melepaskan semua alamat IP privat atau alamat IP Elastis untuk digunakan oleh instans lain.

Anda tidak dapat menghapus antarmuka jaringan yang sedang digunakan. Pertama, Anda harus lepaskan antarmuka jaringan.

Console

Untuk menghapus antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan.
3. Pilih kotak centang untuk antarmuka jaringan, dan kemudian pilih Tindakan, Hapus.
4. Saat diminta konfirmasi, pilih Hapus.

AWS CLI

Untuk menghapus antarmuka jaringan

Gunakan perintah delete-network-interface berikut.

```
aws ec2 delete-network-interface --network-interface-id eni-1234567890abcdef0
```

PowerShell

Untuk menghapus antarmuka jaringan

Gunakan Remove-EC2NetworkInterface cmdlet.

```
Remove-EC2NetworkInterface -NetworkInterfaceId eni-1234567890abcdef0
```

# Bandwidth jaringan instans Amazon EC2

Spesifikasi bandwidth instans berlaku untuk lalu lintas masuk dan keluar untuk instans. Misalnya, jika sebuah instance menentukan bandwidth hingga 10 Gbps, itu berarti ia memiliki bandwidth hingga 10 Gbps untuk lalu lintas masuk dan, secara bersamaan, hingga 10 Gbps untuk lalu lintas keluar. Bandwidth jaringan yang tersedia untuk instans EC2 tergantung pada beberapa faktor, sebagai berikut.

Multi-flow lalu lintas

Ketika lalu lintas melewati gateway internet atau gateway lokal, bandwidth yang tersedia untuk lalu lintas multi-aliran tergantung pada jenis instans.

| Tipe instans | Bandwidth yang tersedia |
|---|---|
| Jenis instans dengan kurang dari 32 vCPU | Terbatas hingga 5 Gbps |
| Jenis instans dengan lebih dari 32 vCPU | Terbatas hingga 50% dari bandwidth yang tersedia untuk jenis instans |
| Instans C8in, C8ine, M8in, M8ine, M8idn, R8in, R8idn | Terbatas pada bandwidth dasar untuk jenis instans |

Single-flow lalu lintas

Ketika instance tidak berada dalam grup penempatan cluster yang sama, bandwidth untuk lalu lintas aliran tunggal dibatasi hingga 5 Gbps. Untuk mengurangi latensi dan meningkatkan bandwidth alur tunggal, cobalah lakukan salah satu hal berikut:

- Gunakan grup penempatan klaster untuk mencapai bandwidth hingga 10 Gbps untuk instans dalam grup penempatan yang sama.
- Siapkan beberapa jalur antara dua titik akhir untuk mencapai bandwidth yang lebih tinggi dengan Multipath TCP (MPTCP).
- Konfigurasikan ENA Express untuk instans yang memenuhi syarat dalam Zona Ketersediaan yang sama untuk mencapai hingga 25 Gbps di antara instans tersebut.

> **Note**
>
> Aliran tunggal dianggap sebagai aliran TCP atau UDP 5-tupel yang unik. Untuk protokol lain yang mengikuti header IP, seperti `GRE` atau`IPsec`, 3 tupel IP sumber, IP tujuan, dan protokol berikutnya digunakan untuk menentukan aliran.

## Bandwidth instans yang tersedia

Bandwidth jaringan yang tersedia dari sebuah instans tergantung pada jumlah vCPU yang dimilikinya. Misalnya, sebuah `m5.8xlarge` instans memiliki 32 vCPU dan 10 Gbps bandwidth jaringan, dan sebuah `m5.16xlarge` instans memiliki 64 vCPU dan 20 Gbps bandwidth jaringan. Namun, instans mungkin tidak mencapai bandwidth ini; misalnya, jika melebihi perizinan jaringan pada tingkat instans, seperti paket per detik atau jumlah koneksi yang dilacak. Berapa banyak bandwidth yang tersedia yang dapat digunakan lalu lintas tergantung pada jumlah vCPU dan tujuan. Misalnya, sebuah `m5.16xlarge` instance memiliki 64 vCPU, sehingga lalu lintas ke instance lain di Wilayah dapat menggunakan bandwidth penuh yang tersedia (20 Gbps). Namun, lalu lintas yang melewati gateway internet atau gateway lokal hanya dapat menggunakan 50% dari bandwidth yang tersedia (10 Gbps).

Biasanya, instans dengan 16 vCPU atau kurang (ukuran `4xlarge` dan lebih kecil) didokumentasikan memiliki “hingga” bandwidth tertentu; misalnya, “hingga 10 Gbps”. Instans ini memiliki bandwidth acuan. Untuk memenuhi permintaan tambahan, mereka dapat menggunakan mekanisme I/O kredit jaringan untuk melampaui bandwidth dasar mereka. Instans dapat menggunakan lonjakan bandwidth untuk waktu yang terbatas, biasanya dari 5 hingga 60 menit, tergantung pada ukuran instans.

Sebuah instance menerima jumlah maksimum I/O kredit jaringan saat peluncuran. Jika instance menghabiskan I/O kredit jaringannya, ia kembali ke bandwidth dasar. Instans yang sedang berjalan mendapatkan I/O kredit jaringan setiap kali menggunakan bandwidth jaringan yang lebih sedikit daripada bandwidth dasarnya. Instans yang dihentikan tidak mendapatkan I/O kredit jaringan. Lonjakan instans adalah upaya terbaik, bahkan ketika instans memiliki kredit yang tersedia, karena lonjakan bandwidth adalah sumber daya bersama.

Ada bucket I/O kredit jaringan terpisah untuk lalu lintas masuk dan keluar.

Performa jaringan dasar dan lonjakan

Panduan Jenis Instans Amazon EC2 menjelaskan kinerja jaringan untuk setiap jenis instans, ditambah bandwidth jaringan dasar yang tersedia untuk instans yang dapat menggunakan bandwidth burst. Untuk informasi selengkapnya, lihat berikut ini:

- Spesifikasi jaringan — Tujuan umum
- Spesifikasi jaringan - Komputasi dioptimalkan
- Spesifikasi jaringan - Memori dioptimalkan
- Spesifikasi jaringan - Penyimpanan dioptimalkan
- Spesifikasi jaringan - Komputasi yang dipercepat
- Spesifikasi jaringan — High-performance komputasi
- Spesifikasi jaringan — Generasi sebelumnya

Atau, Anda dapat menggunakan alat baris perintah untuk mendapatkan informasi ini. Konsol Amazon EC2 tidak menampilkan bandwidth jaringan dasar dari jenis instans.

AWS CLI

Anda dapat menggunakan perintah describe-instance-types untuk menampilkan informasi tentang jenis instance. Contoh berikut menampilkan informasi performa jaringan untuk semua instans C5.

```
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=c5.*" \
    --query "InstanceTypes[].[InstanceType, NetworkInfo.NetworkPerformance,
 NetworkInfo.NetworkCards[0].BaselineBandwidthInGbps] | sort_by(@,&[2])" \
    --output table
```

Berikut ini adalah output contoh. Jika output Anda kehilangan bandwidth dasar, perbarui ke versi terbaru AWS CLI.

```
-----------------------------------------------
|           DescribeInstanceTypes             |
+--------------+--------------------+---------+
|  c5.large    |  Up to 10 Gigabit  |  0.75   |
|  c5.xlarge   |  Up to 10 Gigabit  |  1.25   |
|  c5.2xlarge  |  Up to 10 Gigabit  |  2.5    |
```

```
|  c5.4xlarge  |  Up to 10 Gigabit  |  5.0  |
|  c5.9xlarge  |  12 Gigabit        |  12.0 |
|  c5.12xlarge |  12 Gigabit        |  12.0 |
|  c5.18xlarge |  25 Gigabit        |  25.0 |
|  c5.24xlarge |  25 Gigabit        |  25.0 |
|  c5.metal    |  25 Gigabit        |  25.0 |
+--------------+--------------------+-------+
```

PowerShell

Anda dapat menggunakan Get-EC2InstanceType PowerShell perintah untuk menampilkan informasi tentang jenis instance. Contoh berikut menampilkan informasi performa jaringan untuk semua instans C5.

```
Get-EC2InstanceType -Filter @{Name = "instance-type"; Values = "c5.*" } | `
    Select-Object `
    InstanceType,
    @{Name = 'NetworkPerformance'; Expression =
 {($_.Networkinfo.NetworkCards.NetworkPerformance)}},
    @{Name = 'BaselineBandwidthInGbps'; Expression =
 {($_.Networkinfo.NetworkCards.BaselineBandwidthInGbps)}} | `
Format-Table -AutoSize
```

Berikut ini adalah output contoh.

```
InstanceType NetworkPerformance BaselineBandwidthInGbps
------------ ------------------ ----------------------
c5.4xlarge   Up to 10 Gigabit                     5.00
c5.xlarge    Up to 10 Gigabit                     1.25
c5.12xlarge  12 Gigabit                          12.00
c5.9xlarge   12 Gigabit                          12.00
c5.24xlarge  25 Gigabit                          25.00
c5.metal     25 Gigabit                          25.00
c5.2xlarge   Up to 10 Gigabit                     2.50
c5.large     Up to 10 Gigabit                     0.75
c5.18xlarge  25 Gigabit                          25.00
```

## Konfigurasi pembobotan bandwidth instans EC2

Beberapa jenis instans mendukung pembobotan bandwidth yang dapat dikonfigurasi, di mana Anda dapat memilih bobot bandwidth dasar yang mendukung pemrosesan jaringan atau operasi

EBS. Pengaturan default untuk bandwidth dasar ditentukan oleh jenis instans Anda. Anda dapat mengonfigurasi bobot bandwidth selama peluncuran, atau mengubah pengaturan instans Anda dengan preferensi pembobotan berikut:

- default — Opsi ini menggunakan konfigurasi bandwidth standar untuk jenis instans Anda.
- vpc-1 — Opsi ini meningkatkan bandwidth dasar yang tersedia untuk jaringan dan mengurangi bandwidth dasar untuk operasi EBS.
- ebs-1 — Opsi ini meningkatkan bandwidth dasar yang tersedia untuk operasi EBS, dan mengurangi bandwidth dasar untuk jaringan.

## Pertimbangan pembobotan bandwidth

Berikut ini adalah beberapa pertimbangan yang mungkin memengaruhi strategi pembobotan bandwidth Anda.

- Menyetel preferensi pembobotan bandwidth hanya memengaruhi spesifikasi bandwidth. Spesifikasi paket jaringan per detik (PPS) dan input/output operasi EBS per detik (IOPS) tidak berubah.
- Spesifikasi bandwidth gabungan antara jaringan dan EBS tidak berubah. Saat Anda memilih konfigurasi pembobotan bandwidth, bandwidth dasar yang tersedia untuk opsi yang dipilih meningkat, dan bandwidth dasar untuk opsi yang tersisa dikurangi dengan jumlah absolut yang sama. Untuk semua instance kecuali instans Flex, bandwidth burst yang tersedia tetap sama untuk opsi yang Anda pilih, dan dikurangi untuk opsi yang tersisa. Untuk instans Flex hingga 4xlarge, bandwidth burst tetap tidak berubah. Untuk instans Flex 8xlarge dan lebih besar, bandwidth burst meningkat dengan jumlah yang sama dengan bandwidth dasar.
- Penting untuk memahami bagaimana perubahan dalam alokasi bandwidth dapat memengaruhi I/O kinerja EBS. Untuk instans EC2 yang memiliki `vpc-1` konfigurasi (peningkatan bandwidth jaringan), Anda mungkin mengalami IOPS yang lebih rendah untuk volume EBS jika Anda mencapai batas bandwidth EBS sebelum mencapai batas IOPS. Ini lebih terlihat dengan I/O ukuran yang lebih besar.

  Misalnya, pada jenis instans yang biasanya mendukung 240.000 IOPS dengan I/O ukuran 16 KiB, jika Anda memilih pem `vpc-1` bobotan, itu mungkin mengurangi IOPS yang dapat dicapai karena batas bandwidth dasar EBS yang disesuaikan.

  Saat merencanakan beban kerja Anda, pertimbangkan I/O ukuran dan pola Anda. I/O Ukuran yang lebih kecil cenderung tidak terpengaruh oleh batasan bandwidth, sementara I/O ukuran yang lebih besar atau beban kerja berurutan mungkin melihat lebih banyak dampak dari perubahan

bandwidth. Selalu uji beban kerja spesifik Anda untuk memastikan kinerja optimal dengan konfigurasi yang Anda pilih.

- Spesifikasi bandwidth multi-aliran jaringan untuk lalu lintas yang melewati gateway internet atau gateway lokal disesuaikan hingga 50% dari bandwidth dasar dari opsi yang dikonfigurasi atau 5 Gbps, jika berlaku. Untuk informasi selengkapnya, lihat Bandwidth jaringan instans Amazon EC2.

  Contoh berikut didasarkan pada jenis instance yang memiliki bandwidth dasar default 40 Gbps, dan bandwidth batas default 20 Gbps. Jika Anda memilih pembobotan `vpc-1` bandwidth untuk contoh ini, bandwidth baseline tertimbang berubah menjadi 50 Gbps, dan bandwidth perbatasan berubah menjadi 25 Gbps.
- Fitur ini tersedia di semua Wilayah komersial, selaras dengan ketersediaan dan dukungan instans EC2.
- Fitur ini tidak menambahkan biaya tambahan ke instans EC2 Anda.

## Jenis instans yang didukung untuk pembobotan bandwidth

Jenis instans dalam keluarga instance berikut mendukung pembobotan bandwidth yang dapat dikonfigurasi.

- Tujuan umum:  M8a, M8g, M8gd, M8i, M8id,, M9g, M9gd M8i-flex
- Komputasi dioptimalkan:  C8a, C8g, C8gd, C8i, C8id,, C9g, C9gd C8i-flex
- Memori dioptimalkan:  R8a, R8g, R8gd, R8i, R8id,, R9g, R9gd, X8g, X8aedz, X8i R8i-flex

## Periksa pengaturan bandwidth saat ini

Untuk melihat pengaturan bandwidth saat ini untuk instance Anda, pilih salah satu tab untuk instruksi.

Console

Untuk mendapatkan pengaturan bandwidth untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance yang ingin Anda periksa dari daftar, dan arahkan ke  tab  Jaringan. Pengaturan Anda saat ini ditampilkan di  bidang Bandwidth yang  dikonfigurasi. Amazon EC2 menggunakan pengaturan default untuk jenis instans Anda jika bandwidth tidak disetel ke nilai tertentu.

AWS CLI

Untuk mendapatkan pengaturan bandwidth untuk sebuah instance

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[].Instances[].NetworkPerformanceOptions.BandwidthWeighting
 \
    --output text
```

Berikut ini adalah output contoh.

```
default
```

Contoh ini mencantumkan semua instans yang memiliki preferensi pembobotan bandwidth disetel kevpc-1, untuk bandwidth jaringan yang lebih tinggi.

```
aws ec2 describe-instances \
    --filters "Name=network-performance-options.bandwidth-weighting,Values=vpc-1" \
    --query Reservations[].Instances[].InstanceId \
    --output text
```

PowerShell

Untuk mendapatkan pengaturan bandwidth untuk sebuah instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance `
    -
InstanceId i-1234567890abcdef0).Instances.NetworkPerformanceOptions.BandwidthWeighting.Value
```

Berikut ini adalah output contoh.

```
default
```

Contoh ini mencantumkan semua instans yang memiliki preferensi pembobotan bandwidth disetel kevpc-1, untuk bandwidth jaringan yang lebih tinggi.

```
(Get-EC2Instance `
```

```
    -Filter @{Name="network-performance-options.bandwidth-
weighting";Values="vpc-1"}).Instances.InstanceId
```

## Konfigurasikan bobot bandwidth untuk instans Anda

Anda dapat mengonfigurasi bobot bandwidth baik saat peluncuran atau dengan memodifikasi instans yang ada dari konsol EC2, API/SDKs atau CLI.

Mengkonfigurasi bobot bandwidth saat Anda meluncurkan instance

Untuk mengkonfigurasi pengaturan bandwidth saat Anda meluncurkan instance, pilih salah satu tab untuk instruksi.

Anda juga dapat menentukan bobot bandwidth dalam template peluncuran. Untuk membuat template peluncuran, lihatBuat template peluncuran Amazon EC2. Parameter yang akan diatur berada di lokasi yang sama seperti untuk meluncurkan instance langsung dari konsol. Perluas bagian Detail lanjutan, dan atur konfigurasi bandwidth Instance.

Untuk meluncurkan instance dengan template peluncuran Anda, lihatLuncurkan instans EC2 menggunakan template peluncuran.

Console

Untuk meluncurkan instance dengan pembobotan bandwidth yang dapat dikonfigurasi

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih Luncurkan Instans. Ini membuka dialog Luncurkan instance. Ada beberapa cara tambahan yang bisa Anda dapatkan ke dialog peluncuran, tergantung pada preferensi Anda. Misalnya, Anda dapat meluncurkan instans langsung dari AMI atau dari dasbor Amazon EC2 itu sendiri.
4. Amazon Machine Image (AMI) yang Anda luncurkan harus didasarkan pada `Arm` arsitektur. Banyak gambar Mulai Cepat mendukung keduanya `x86` dan `Arm` arsitektur, Setelah Anda memilih sistem operasi untuk instance Anda, pilih `Arm` opsi dari daftar Arsitektur.
5. Jenis instance harus salah satu Tipe instans yang didukung untuk fitur ini.
6. Saat Anda memperluas bagian Detail lanjutan, Anda dapat menggulir ke bawah untuk menemukan pengaturan konfigurasi bandwidth Instance. Pilih opsi konfigurasi bandwidth untuk instance Anda.

7. Konfigurasikan semua pengaturan lain untuk instance Anda seperti biasanya, dan pilih Luncurkan instance.

AWS CLI

Untuk meluncurkan instance dengan pembobotan bandwidth yang dapat dikonfigurasi

Gunakan perintah run-instance dengan opsi berikut untuk meluncurkan instans yang dikonfigurasi untuk bobot bandwidth jaringan yang lebih tinggi.

```
--network-performance-options BandwidthWeighting=vpc-1
```

Gunakan perintah run-instance dengan opsi berikut untuk meluncurkan instans yang dikonfigurasi untuk bobot bandwidth EBS yang lebih tinggi.

```
--network-performance-options BandwidthWeighting=ebs-1
```

PowerShell

Untuk meluncurkan instance dengan pembobotan bandwidth yang dapat dikonfigurasi

Gunakan New-EC2Instance cmdlet dengan parameter berikut untuk meluncurkan instance yang dikonfigurasi untuk bobot bandwidth jaringan yang lebih tinggi.

```
-NetworkPerformanceOptions_BandwidthWeighting vpc-1
```

Gunakan New-EC2Instance cmdlet dengan parameter berikut untuk meluncurkan instans yang dikonfigurasi untuk bobot bandwidth EBS yang lebih tinggi.

```
-NetworkPerformanceOptions_BandwidthWeighting ebs-1
```

Memperbarui bobot bandwidth untuk instance yang ada

Untuk memperbarui bobot bandwidth untuk instans yang sudah ada, instans Anda harus berada dalam `Stopped` status tersebut.

### Console

Untuk memperbarui pembobotan bandwidth

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance yang ingin Anda perbarui dari daftar.
4. Sebelum Anda mengubah konfigurasi bandwidth, instans Anda harus dalam `Stopped` status. Jika instans Anda sedang berjalan, pilih Hentikan instance dari menu Status Instance.
5. Pilih Kelola bandwidth dari menu Tindakan > Jaringan. Ini membuka dialog Kelola bandwidth.

> **Note**
>
> Jika jenis instans Anda tidak mendukung konfigurasi untuk pembobotan bandwidth, item menu tersebut dinonaktifkan.

6. Pilih opsi untuk memperbarui instance Anda, dan pilih U bah untuk menyimpan pengaturan Anda.

### AWS CLI

Untuk memperbarui pembobotan bandwidth

Gunakan perintah modify-instance-network-performance-options untuk mengonfigurasi bobot bandwidth jaringan yang lebih tinggi untuk instance yang ditentukan.

```
aws ec2 modify-instance-network-performance-options \
    --instance-id i-1234567890abcdef0 \
    --bandwidth-weighting=vpc-1
```

Contoh berikut mengonfigurasi bobot bandwidth EBS yang lebih tinggi untuk instance yang ditentukan.

```
aws ec2 modify-instance-network-performance-options \
    --instance-id i-1234567890abcdef0 \
    --bandwidth-weighting=ebs-1
```

PowerShell

Untuk memperbarui pembobotan bandwidth

Gunakan Edit-EC2InstanceNetworkPerformanceOption cmdlet untuk mengonfigurasi bobot bandwidth jaringan yang lebih tinggi untuk instance yang ditentukan.

```
Edit-EC2InstanceNetworkPerformanceOption `
    -InstanceId i-1234567890abcdef0 `
    -BandwidthWeighting vpc-1
```

Contoh berikut mengonfigurasi bobot bandwidth EBS yang lebih tinggi untuk instance yang ditentukan.

```
Edit-EC2InstanceNetworkPerformanceOption `
    -InstanceId i-1234567890abcdef0 `
    -BandwidthWeighting ebs-1
```

## Kinerja jaringan

Tabel berikut menunjukkan kinerja jaringan, dalam Gbps, yang dapat dicapai dengan`default`,`vpc-1`, dan `ebs-1` konfigurasi.

| Tipe instans | **`default`** (Dasar/Burst) | **`vpc-1`** (Dasar/Burst) | **`ebs-1`** (Dasar/Burst) |
|---|---|---|---|
| c8a.medium | 0, 52/12, 5 | 0, 65/12, 5 | 0.438/10.0 |
| c8a.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| c8a.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10,0 |
| c8a.2xbesar | 3.75/15.0 | 4.688/15.0 | 3.125/12.5 |
| c8a.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| c8a.8xbesar | 15.0 | 18.75 | 12.5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8a.12xbesar | 22.5 | 28.125 | 18.75 |
| c8a.16xbesar | 30.0 | 37,5 | 25.0 |
| c8a.24xbesar | 40,0 | 50,0 | 32.5 |
| c8a.48xbesar | 75.0 | 93.75 | 60.0 |
| c8a.logam-24xl | 40,0 | 50,0 | 32.5 |
| c8a.logam-48xl | 75.0 | 93.75 | 60.0 |
| c8g.sedang | 0, 52/12, 5 | 0, 65/12, 5 | 0.441/10.0 |
| c8g.besar | 0.937/12.5 | 1.171/1.5 | 0.779/10.0 |
| c8g.xlarge | 1,875/12,5 | 2.344/ 12.5 | 1.562/10,0 |
| c8g.2xbesar | 3.75/15.0 | 4.688/15.0 | 3.125/12.5 |
| c8g.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| c8g.8xbesar | 15.0 | 18.75 | 12.5 |
| c8g.12xbesar | 22.5 | 28.125 | 18.75 |
| c8g.16xbesar | 30.0 | 37,5 | 25.0 |
| c8g.24xbesar | 40,0 | 50,0 | 32.5 |
| c8g.48xbesar | 50,0 | 62.5 | 40,0 |
| c8g.logam-24xl | 40,0 | 50,0 | 32.5 |
| c8g.logam-48xl | 50,0 | 62.5 | 40,0 |
| c8gd.medium | 0, 52/12, 5 | 0, 65/12, 5 | 0.441/10.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8gd.besar | 0.937/12.5 | 1.171/1.5 | 0.779/10.0 |
| c8gd.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10,0 |
| c8gd.2xbesar | 3.75/15,0 | 4.688/15.0 | 3.125/12.5 |
| c8gd.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| c8gd.8xbesar | 15.0 | 18.75 | 12.5 |
| c8gd.12xbesar | 22.5 | 28.125 | 18.75 |
| c8gd.16xbesar | 30.0 | 37,5 | 25.0 |
| c8gd.24xbesar | 40,0 | 50,0 | 32.5 |
| c8gd.48xbesar | 50,0 | 62.5 | 40,0 |
| c8gd.logam-24xl | 40,0 | 50,0 | 32.5 |
| c8gd.logam-48xl | 50,0 | 62.5 | 40,0 |
| c8i.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| c8i.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10,0 |
| c8i.2xbesar | 3.75/15,0 | 4.688/15.0 | 3.125/12.5 |
| c8i.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| c8i.8xbesar | 15.0 | 18.75 | 12.5 |
| c8i.12xbesar | 22.5 | 28.125 | 18.75 |
| c8i.16xbesar | 30.0 | 37,5 | 25.0 |
| c8i.24xbesar | 40,0 | 50,0 | 32.5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8i.32xbesar | 50,0 | 62.5 | 40,0 |
| c8i.48xbesar | 75.0 | 93.75 | 60.0 |
| c8i.96xbesar | 100.0 | 125.0 | 80.0 |
| c8i.logam-48xl | 75.0 | 93.75 | 60.0 |
| c8i.logam-96xl | 100.0 | 125.0 | 80.0 |
| c8id.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| c8id.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10,0 |
| c8id.2xbesar | 3.75/15.0 | 4.688/15.0 | 3.125/12.5 |
| c8id.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| c8id.8xbesar | 15.0 | 18.75 | 12.5 |
| c8id.12xbesar | 22.5 | 28.125 | 18.75 |
| c8id.16xbesar | 30.0 | 37,5 | 25.0 |
| c8id.24xbesar | 40,0 | 50,0 | 32.5 |
| c8id.32xbesar | 50,0 | 62.5 | 40,0 |
| c8id.48xbesar | 75.0 | 93.75 | 60.0 |
| c8id.96xbesar | 100.0 | 125.0 | 80.0 |
| c8id.logam-48xl | 75.0 | 93.75 | 60.0 |
| c8id.logam-96xl | 100.0 | 125.0 | 80.0 |
| c8i-flex.besar | 0.468/ 12.5 | 0,585/12,5 | 0.389/10.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8i-flex.xlarge | 0.937/12.5 | 1.172/1.5 | 0.779/10.0 |
| c8i-flex.2xbesar | 1.875/15.0 | 2.344/15.0 | 1.562/12.5 |
| c8i-flex.4xbesar | 3.75/15.0 | 4.688/15.0 | 3.125/12.5 |
| c8i-flex.8xbesar | 7.5/15.0 | 9.375/18.75 | 6, 25/12, 5 |
| c8i-flex.12xbesar | 11, 25/22, 5 | 14.063/28.125 | 9.375/18.75 |
| c8i-flex.16xbesar | 15.0/30.0 | 18, 75/37, 5 | 12.5/25.0 |
| c9g.sedang | 0.55/15.0 | 0,688/15,0 | 0.455/12.0 |
| c9g.besar | 1.0/15.0 | 1,25/15,0 | 0.81/12.0 |
| c9g.xlarge | 2.1/15.0 | 2.625/15.0 | 1.725/12.0 |
| c9g.2xbesar | 4,25/17,0 | 5.313/17.0 | 3,5/14,0 |
| c9g.4xbesar | 8.5/17.0 | 10.625/17.0 | 7.0/14.0 |
| c9g.8xbesar | 17.0 | 21.25 | 14.0 |
| c9g.12xbesar | 25.0 | 31.25 | 20.5 |
| c9g.16xbesar | 34.0 | 42.5 | 28.0 |
| c9g.24xbesar | 50,0 | 62.5 | 41.0 |
| c9g.48xbesar | 100.0 | 125.0 | 82.0 |
| c9g.logam-48xl | 100.0 | 125.0 | 82.0 |
| c9gd.medium | 0.55/15.0 | 0,688/15,0 | 0.455/12.0 |
| c9gd.large | 1.0/15.0 | 1,25/15,0 | 0.81/12.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c9gd.xlarge | 2.1/15.0 | 2.625/15.0 | 1.725/12.0 |
| c9gd.2xbesar | 4,25/17,0 | 5.313/17.0 | 3,5/14,0 |
| c9gd.4xbesar | 8.5/17.0 | 10.625/17.0 | 7.0/14.0 |
| c9gd.8xbesar | 17.0 | 21.25 | 14.0 |
| c9gd.12xbesar | 25.0 | 31.25 | 20.5 |
| c9gd.16xbesar | 34.0 | 42.5 | 28.0 |
| c9gd.24xbesar | 50,0 | 62.5 | 41.0 |
| c9gd.48xbesar | 100.0 | 125.0 | 82.0 |
| c9gd.logam-48xl | 100.0 | 125.0 | 82.0 |
| m8a.medium | 0, 52/12, 5 | 0, 65/12, 5 | 0.438/10.0 |
| m8a.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| m8a.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| m8a.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| m8a.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| m8a.8xbesar | 15.0 | 18.75 | 12.5 |
| m8a.12xbesar | 22.5 | 28.125 | 18.75 |
| m8a.16xbesar | 30.0 | 37,5 | 25.0 |
| m8a.24xbesar | 40,0 | 50,0 | 32.5 |
| m8a.48xbesar | 75.0 | 93.75 | 60.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8a.logam-24xl | 40,0 | 50,0 | 32.5 |
| m8a.logam-48xl | 75.0 | 93.75 | 60.0 |
| m8g.sedang | 0, 52/12, 5 | 0, 65/12, 5 | 0.441/10.0 |
| m8g.besar | 0.937/12.5 | 1.171/1.5 | 0.779/10.0 |
| m8g.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| m8g.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| m8g.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| m8g.8xbesar | 15.0 | 18.75 | 12.5 |
| m8g.12xbesar | 22.5 | 28.125 | 18.75 |
| m8g.16xbesar | 30.0 | 37,5 | 25.0 |
| m8g.24xbesar | 40,0 | 50,0 | 32.5 |
| m8g.48xbesar | 50,0 | 62.5 | 40,0 |
| m8g.logam-24xl | 40,0 | 50,0 | 32.5 |
| m8g.logam-48xl | 50,0 | 62.5 | 40,0 |
| m8gd.medium | 0, 52/12, 5 | 0, 65/12, 5 | 0.441/10.0 |
| m8gd.besar | 0.937/12.5 | 1.171/1.5 | 0.779/10.0 |
| m8gd.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| m8gd.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| m8gd.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8gd.8xbesar | 15.0 | 18.75 | 12.5 |
| m8gd.12xbesar | 22.5 | 28.125 | 18.75 |
| m8gd.16xbesar | 30.0 | 37,5 | 25.0 |
| m8gd.24xbesar | 40,0 | 50,0 | 32.5 |
| m8gd.48xbesar | 50,0 | 62.5 | 40,0 |
| m8gd.logam-24xl | 40,0 | 50,0 | 32.5 |
| m8gd.logam-48xl | 50,0 | 62.5 | 40,0 |
| m8i.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| m8i.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| m8i.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| m8i.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| m8i.8xbesar | 15.0 | 18.75 | 12.5 |
| m8i.12xbesar | 22.5 | 28.125 | 18.75 |
| m8i.16xbesar | 30.0 | 37,5 | 25.0 |
| m8i.24xbesar | 40,0 | 50,0 | 32.5 |
| m8i.32xbesar | 50,0 | 62.5 | 40,0 |
| m8i.48xbesar | 75.0 | 93.75 | 60.0 |
| m8i.96xbesar | 100.0 | 125.0 | 80.0 |
| m8i.logam-48xl | 75.0 | 93.75 | 60.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8i.logam-96xl | 100.0 | 125.0 | 80.0 |
| m8id.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| m8id.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| m8id.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| m8id.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| m8id.8xbesar | 15.0 | 18.75 | 12.5 |
| m8id.12xbesar | 22.5 | 28.125 | 18.75 |
| m8id.16xbesar | 30.0 | 37,5 | 25.0 |
| m8id.24xbesar | 40,0 | 50,0 | 32.5 |
| m8id.32xbesar | 50,0 | 62.5 | 40,0 |
| m8id.48xbesar | 75.0 | 93.75 | 60.0 |
| m8id.96xbesar | 100.0 | 125.0 | 80.0 |
| m8id.logam-48xl | 75.0 | 93.75 | 60.0 |
| m8id.logam-96xl | 100.0 | 125.0 | 80.0 |
| m8i-flex.besar | 0.468/ 12.5 | 0,585/12,5 | 0.389/10.0 |
| m8i-flex.xlarge | 0.937/12.5 | 1.172/1.5 | 0.779/10.0 |
| m8i-flex.2xbesar | 1.875/15.0 | 2.344/15.0 | 1.562/12.5 |
| m8i-flex.4xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| m8i-flex.8xbesar | 7.5/15.0 | 9.375/18.75 | 6, 25/12, 5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8i-flex.12xbesar | 11, 25/22, 5 | 14.063/28.125 | 9.375/18.75 |
| m8i-flex.16xbesar | 15.0/30.0 | 18, 75/37, 5 | 12.5/25.0 |
| m9g.sedang | 0.55/15.0 | 0,688/15,0 | 0.455/12.0 |
| m9g.besar | 1.0/15.0 | 1,25/15,0 | 0.81/12.0 |
| m9g.xlarge | 2.1/15.0 | 2.625/15.0 | 1.725/12.0 |
| m9g.2xbesar | 4,25/17,0 | 5.313/17.0 | 3,5/14,0 |
| m9g.4xbesar | 8.5/17.0 | 10.625/17.0 | 7.0/14.0 |
| m9g.8xbesar | 17.0 | 21.25 | 14.0 |
| m9g.12xbesar | 25.0 | 31.25 | 20.5 |
| m9g.16xbesar | 34.0 | 42.5 | 28.0 |
| m9g.24xbesar | 50,0 | 62.5 | 41.0 |
| m9g.48xbesar | 100.0 | 125.0 | 82.0 |
| m9g.logam-48xl | 100.0 | 125.0 | 82.0 |
| m9gd.medium | 0.55/15.0 | 0,688/15,0 | 0.455/12.0 |
| m9gd.besar | 1.0/15.0 | 1,25/15,0 | 0.81/12.0 |
| m9gd.xlarge | 2.1/15.0 | 2.625/15.0 | 1.725/12.0 |
| m9gd.2xbesar | 4,25/17,0 | 5.313/17.0 | 3,5/14,0 |
| m9gd.4xbesar | 8.5/17.0 | 10.625/17.0 | 7.0/14.0 |
| m9gd.8xbesar | 17.0 | 21.25 | 14.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m9gd.12xbesar | 25.0 | 31.25 | 20.5 |
| m9gd.16xbesar | 34.0 | 42.5 | 28.0 |
| m9gd.24xbesar | 50,0 | 62.5 | 41.0 |
| m9gd.48xbesar | 100.0 | 125.0 | 82.0 |
| m9gd.logam-48xl | 100.0 | 125.0 | 82.0 |
| r8a.medium | 0, 52/12, 5 | 0, 65/12, 5 | 0.438/10.0 |
| r8a.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| r8a.xlarge | 1,875/12,5 | 2.344/ 12.5 | 1.562/10.0 |
| r8a.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| r8a.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| r8a.8xbesar | 15.0 | 18.75 | 12.5 |
| r8a.12xbesar | 22.5 | 28.125 | 18.75 |
| r8a.16xbesar | 30.0 | 37,5 | 25.0 |
| r8a.24xbesar | 40,0 | 50,0 | 32.5 |
| r8a.48xbesar | 75.0 | 93.75 | 60.0 |
| r8a.logam-24xl | 40,0 | 50,0 | 32.5 |
| r8a.logam-48xl | 75.0 | 93.75 | 60.0 |
| r8g.sedang | 0, 52/12, 5 | 0, 65/12, 5 | 0.441/10.0 |
| r8g.besar | 0.937/12.5 | 1.171/1.5 | 0.779/10.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r8g.xbesar | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| r8g.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| r8g.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| r8g.8xbesar | 15.0 | 18.75 | 12.5 |
| r8g.12xbesar | 22.5 | 28.125 | 18.75 |
| r8g.16xbesar | 30.0 | 37,5 | 25.0 |
| r8g.24xbesar | 40,0 | 50,0 | 32.5 |
| r8g.48xbesar | 50,0 | 62.5 | 40,0 |
| r8g.logam-24xl | 40,0 | 50,0 | 32.5 |
| r8g.logam-48xl | 50,0 | 62.5 | 40,0 |
| r8gd.medium | 0, 52/12, 5 | 0, 65/12, 5 | 0.441/10.0 |
| r8gd.besar | 0.937/12.5 | 1.171/1.5 | 0.779/10.0 |
| r8gd.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| r8gd.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| r8gd.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| r8gd.8xbesar | 15.0 | 18.75 | 12.5 |
| r8gd.12xbesar | 22.5 | 28.125 | 18.75 |
| r8gd.16xbesar | 30.0 | 37,5 | 25.0 |
| r8gd.24xbesar | 40,0 | 50,0 | 32.5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r8gd.48xbesar | 50,0 | 62.5 | 40,0 |
| r8gd.logam-24xl | 40,0 | 50,0 | 32.5 |
| r8gd.logam-48xl | 50,0 | 62.5 | 40,0 |
| r8i.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| r8i.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| r8i.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| r8i.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| r8i.8xbesar | 15.0 | 18.75 | 12.5 |
| r8i.12xbesar | 22.5 | 28.125 | 18.75 |
| r8i.16xbesar | 30.0 | 37,5 | 25.0 |
| r8i.24xbesar | 40,0 | 50,0 | 32.5 |
| r8i.32xbesar | 50,0 | 62.5 | 40,0 |
| r8i.48xbesar | 75.0 | 93.75 | 60.0 |
| r8i.96xbesar | 100.0 | 125.0 | 80.0 |
| r8i.logam-48xl | 75.0 | 93.75 | 60.0 |
| r8i.logam-96xl | 100.0 | 125.0 | 80.0 |
| r8id.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| r8id.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| r8id.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r8id.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| r8id.8xbesar | 15.0 | 18.75 | 12.5 |
| r8id.12xbesar | 22.5 | 28.125 | 18.75 |
| r8id.16xbesar | 30.0 | 37,5 | 25.0 |
| r8id.24xbesar | 40,0 | 50,0 | 32.5 |
| r8id.32xbesar | 50,0 | 62.5 | 40,0 |
| r8id.48xbesar | 75.0 | 93.75 | 60.0 |
| r8id.96xbesar | 100.0 | 125.0 | 80.0 |
| r8id.logam-48xl | 75.0 | 93.75 | 60.0 |
| r8id.logam-96xl | 100.0 | 125.0 | 80.0 |
| r8i-flex.besar | 0.468/ 12.5 | 0,585/12,5 | 0.389/10.0 |
| r8i-flex.xlarge | 0.937/12.5 | 1.172/1.5 | 0.779/10.0 |
| r8i-flex.2xbesar | 1.875/15.0 | 2.344/15.0 | 1.562/12.5 |
| r8i-flex.4xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| r8i-flex.8xbesar | 7.5/15.0 | 9.375/18.75 | 6, 25/12, 5 |
| r8i-flex.12xbesar | 11, 25/22, 5 | 14.063/28.125 | 9.375/18.75 |
| r8i-flex.16xbesar | 15.0/30.0 | 18, 75/37, 5 | 12.5/25.0 |
| r9g.sedang | 0.55/15.0 | 0,688/15,0 | 0.455/12.0 |
| r9g.besar | 1.0/15.0 | 1,25/15,0 | 0.81/12.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r9g.xbesar | 2.1/15.0 | 2.625/15.0 | 1.725/12.0 |
| r9g.2xbesar | 4,25/17,0 | 5.313/17.0 | 3,5/14,0 |
| r9g.4xbesar | 8.5/17.0 | 10.625/17.0 | 7.0/14.0 |
| r9g.8xbesar | 17.0 | 21.25 | 14.0 |
| r9g.12xbesar | 25.0 | 31.25 | 20.5 |
| r9g.16xbesar | 34.0 | 42.5 | 28.0 |
| r9g.24xbesar | 50,0 | 62.5 | 41.0 |
| r9g.48xbesar | 100.0 | 125.0 | 82.0 |
| r9g.logam-48xl | 100.0 | 125.0 | 82.0 |
| r9gd.medium | 0.55/15.0 | 0,688/15,0 | 0.455/12.0 |
| r9gd.besar | 1.0/15.0 | 1,25/15,0 | 0.81/12.0 |
| r9gd.xlarge | 2.1/15.0 | 2.625/15.0 | 1.725/12.0 |
| r9gd.2xbesar | 4,25/17,0 | 5.313/17.0 | 3,5/14,0 |
| r9gd.4xbesar | 8.5/17.0 | 10.625/17.0 | 7.0/14.0 |
| r9gd.8xbesar | 17.0 | 21.25 | 14.0 |
| r9gd.12xbesar | 25.0 | 31.25 | 20.5 |
| r9gd.16xbesar | 34.0 | 42.5 | 28.0 |
| r9gd.24xbesar | 50,0 | 62.5 | 41.0 |
| r9gd.48xbesar | 100.0 | 125.0 | 82.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r9gd.logam-48xl | 100.0 | 125.0 | 82.0 |
| x8g.sedang | 0, 52/12, 5 | 0, 65/12, 5 | 0.441/10.0 |
| x8g.besar | 0.937/12.5 | 1.171/1.5 | 0.779/10.0 |
| x8g.xbesar | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| x8g.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| x8g.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| x8g.8xbesar | 15.0 | 18.75 | 12.5 |
| x8g.12xbesar | 22.5 | 28.125 | 18.75 |
| x8g.16xbesar | 30.0 | 37,5 | 25.0 |
| x8g.24xbesar | 40,0 | 50,0 | 32.5 |
| x8g.48xbesar | 50,0 | 62.5 | 40,0 |
| x8g.logam-24xl | 40,0 | 50,0 | 32.5 |
| x8g.logam-48xl | 50,0 | 62.5 | 40,0 |
| x8aedz.large | 1.562/18.75 | 1.953/18.75 | 1.249/15.0 |
| x8aedz.xlarge | 3.125/18.75 | 3.907/18.75 | 2.5/15.0 |
| x8aedz.3xbesar | 9.375/18.75 | 11.719/18.75 | 7.5/15.0 |
| x8aedz.6xbesar | 18.75 | 23.438 | 15.0 |
| x8aedz.12xbesar | 37,5 | 46.875 | 30.0 |
| x8aedz.24xbesar | 75.0 | 93.75 | 60.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| x8aedz.logam-12xl | 37,5 | 46.875 | 30.0 |
| x8aedz.logam-24xl | 75.0 | 93.75 | 60.0 |
| x8i.besar | 0.937/12.5 | 1.172/1.5 | 0.774/10.0 |
| x8i.xlarge | 1.875/12.5 | 2.344/ 12.5 | 1.562/10.0 |
| x8i.2xbesar | 3,75/15,0 | 4.688/15.0 | 3.125/12.5 |
| x8i.4xbesar | 7.5/15.0 | 9.375/15.0 | 6, 25/12, 5 |
| x8i.8xbesar | 15.0 | 18.75 | 12.5 |
| x8i.12xbesar | 22.5 | 28.125 | 18.75 |
| x8i.16xbesar | 30.0 | 37,5 | 25.0 |
| x8i.24xbesar | 40,0 | 50,0 | 32.5 |
| x8i.32xbesar | 50,0 | 62.5 | 40,0 |
| x8i.48xbesar | 75.0 | 93.75 | 60.0 |
| x8i.64xbesar | 80.0 | 100.0 | 62.5 |
| x8i.96xbesar | 100.0 | 125.0 | 80.0 |
| x8i.logam-48xl | 75.0 | 93.75 | 60.0 |
| x8i.logam-96xl | 100.0 | 125.0 | 80.0 |

## Kinerja Amazon EBS

Tabel berikut menunjukkan kinerja Amazon EBS, dalam Gbps, yang dapat dicapai dengan`default`,`vpc-1`, dan `ebs-1` konfigurasi.

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8a.medium | 0.325/10,0 | 0.195/6.25 | 0.407/10.0 |
| c8a.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| c8a.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10,0 |
| c8a.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| c8a.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| c8a.8xbesar | 10.0 | 6.25 | 12.5 |
| c8a.12xbesar | 15.0 | 9.375 | 18.75 |
| c8a.16xbesar | 20.0 | 12.5 | 25.0 |
| c8a.24xbesar | 30.0 | 20.0 | 37,5 |
| c8a.48xbesar | 60.0 | 41.25 | 75.0 |
| c8a.logam-24xl | 30.0 | 20.0 | 37,5 |
| c8a.logam-48xl | 60.0 | 41.25 | 75.0 |
| c8g.sedang | 0.315/10,0 | 0.185/6.25 | 0.394/10.0 |
| c8g.besar | 0,63/10,0 | 0.396/6.25 | 0.788/10.0 |
| c8g.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| c8g.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| c8g.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| c8g.8xbesar | 10.0 | 6.25 | 12.5 |
| c8g.12xbesar | 15.0 | 9.375 | 18.75 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8g.16xbesar | 20.0 | 12.5 | 25.0 |
| c8g.24xbesar | 30.0 | 20.0 | 37,5 |
| c8g.48xbesar | 40,0 | 27.5 | 50,0 |
| c8g.logam-24xl | 30.0 | 20.0 | 37,5 |
| c8g.logam-48xl | 40,0 | 27.5 | 50,0 |
| c8gd.medium | 0.315/10,0 | 0.185/6.25 | 0.394/10.0 |
| c8gd.besar | 0,63/10,0 | 0.396/6.25 | 0.788/10.0 |
| c8gd.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| c8gd.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| c8gd.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| c8gd.8xbesar | 10.0 | 6.25 | 12.5 |
| c8gd.12xbesar | 15.0 | 9.375 | 18.75 |
| c8gd.16xbesar | 20.0 | 12.5 | 25.0 |
| c8gd.24xbesar | 30.0 | 20.0 | 37,5 |
| c8gd.48xbesar | 40,0 | 27.5 | 50,0 |
| c8gd.logam-24xl | 30.0 | 20.0 | 37,5 |
| c8gd.logam-48xl | 40,0 | 27.5 | 50,0 |
| c8i.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| c8i.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8i.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| c8i.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| c8i.8xbesar | 10.0 | 6.25 | 12.5 |
| c8i.12xbesar | 15.0 | 9.375 | 18.75 |
| c8i.16xbesar | 20.0 | 12.5 | 25.0 |
| c8i.24xbesar | 30.0 | 20.0 | 37,5 |
| c8i.32xbesar | 40,0 | 27.5 | 50,0 |
| c8i.48xbesar | 60.0 | 41.25 | 75.0 |
| c8i.96xbesar | 80.0 | 55.0 | 100.0 |
| c8i.logam-48xl | 60.0 | 41.25 | 75.0 |
| c8i.logam-96xl | 80.0 | 55.0 | 100.0 |
| c8id.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| c8id.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| c8id.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| c8id.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| c8id.8xbesar | 10.0 | 6.25 | 12.5 |
| c8id.12xbesar | 15.0 | 9.375 | 18.75 |
| c8id.16xbesar | 20.0 | 12.5 | 25.0 |
| c8id.24xbesar | 30.0 | 20.0 | 37,5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c8id.32xbesar | 40,0 | 27.5 | 50,0 |
| c8id.48xbesar | 60.0 | 41.25 | 75.0 |
| c8id.96xbesar | 80.0 | 55.0 | 100.0 |
| c8id.logam-48xl | 60.0 | 41.25 | 75.0 |
| c8id.logam-96xl | 80.0 | 55.0 | 100.0 |
| c8i-flex.besar | 0.315/10,0 | 0.198/6.25 | 0.394/10.0 |
| c8i-flex.xlarge | 0,63/10,0 | 0.395/6.25 | 0.788/10.0 |
| c8i-flex.2xbesar | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| c8i-flex.4xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| c8i-flex.8xbesar | 5.0/10.0 | 3.125/6.25 | 6, 25/12, 5 |
| c8i-flex.12xbesar | 7.5/15.0 | 4.687/9.375 | 9.375/18.75 |
| c8i-flex.16xbesar | 10.0/20.0 | 6, 25/12, 5 | 12.5/25.0 |
| c9g.sedang | 0.38/12.0 | 0.242/7.75 | 0.475/12.0 |
| c9g.besar | 0.76/12.0 | 0.51/7.75 | 0.95/ 12.0 |
| c9g.xlarge | 1.5/12.0 | 0.975/7.75 | 1.875/12.0 |
| c9g.2xbesar | 3,0/12,0 | 1.937/7.75 | 3.75/12.0 |
| c9g.4xbesar | 6, 0/12, 0 | 3.875/7.75 | 7.5/12.0 |
| c9g.8xbesar | 12.0 | 7.75 | 15.0 |
| c9g.12xbesar | 18.0 | 11.75 | 22.5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| c9g.16xbesar | 24.0 | 15.5 | 30.0 |
| c9g.24xbesar | 36.0 | 23.5 | 45.0 |
| c9g.48xbesar | 72.0 | 47.0 | 90.0 |
| c9g.logam-48xl | 72.0 | 47.0 | 90.0 |
| c9gd.medium | 0.38/12.0 | 0.242/7.75 | 0.475/12.0 |
| c9gd.large | 0.76/12.0 | 0.51/7.75 | 0.95/ 12.0 |
| c9gd.xlarge | 1.5/12.0 | 0.975/7.75 | 1.875/12.0 |
| c9gd.2xbesar | 3,0/12,0 | 1.937/7.75 | 3.75/12.0 |
| c9gd.4xbesar | 6.0/12.0 | 3.875/7.75 | 7.5/12.0 |
| c9gd.8xbesar | 12.0 | 7.75 | 15.0 |
| c9gd.12xbesar | 18.0 | 11.75 | 22.5 |
| c9gd.16xbesar | 24.0 | 15.5 | 30.0 |
| c9gd.24xbesar | 36.0 | 23.5 | 45.0 |
| c9gd.48xbesar | 72.0 | 47.0 | 90.0 |
| c9gd.logam-48xl | 72.0 | 47.0 | 90.0 |
| m8a.medium | 0.325/10,0 | 0.195/6.25 | 0.407/10.0 |
| m8a.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| m8a.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| m8a.2xbesar | 2.5/10.0 | 1.562/6.25 | 3.125/10.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8a.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| m8a.8xbesar | 10.0 | 6.25 | 12.5 |
| m8a.12xbesar | 15.0 | 9.375 | 18.75 |
| m8a.16xbesar | 20.0 | 12.5 | 25.0 |
| m8a.24xbesar | 30.0 | 20.0 | 37,5 |
| m8a.48xbesar | 60.0 | 41.25 | 75.0 |
| m8a.logam-24xl | 30.0 | 20.0 | 37,5 |
| m8a.logam-48xl | 60.0 | 41.25 | 75.0 |
| m8g.sedang | 0.315/10.0 | 0.185/6.25 | 0.394/10.0 |
| m8g.besar | 0,63/10,0 | 0.396/6.25 | 0.788/10.0 |
| m8g.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| m8g.2xbesar | 2.5/10.0 | 1.562/6.25 | 3.125/10.0 |
| m8g.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| m8g.8xbesar | 10.0 | 6.25 | 12.5 |
| m8g.12xbesar | 15.0 | 9.375 | 18.75 |
| m8g.16xbesar | 20.0 | 12.5 | 25.0 |
| m8g.24xbesar | 30.0 | 20.0 | 37,5 |
| m8g.48xbesar | 40,0 | 27.5 | 50,0 |
| m8g.logam-24xl | 30.0 | 20.0 | 37,5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8g.logam-48xl | 40,0 | 27.5 | 50,0 |
| m8gd.medium | 0.315/10.0 | 0.185/6.25 | 0.394/10.0 |
| m8gd.besar | 0,63/10,0 | 0.396/6.25 | 0.788/10.0 |
| m8gd.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| m8gd.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| m8gd.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| m8gd.8xbesar | 10.0 | 6.25 | 12.5 |
| m8gd.12xbesar | 15.0 | 9.375 | 18.75 |
| m8gd.16xbesar | 20.0 | 12.5 | 25.0 |
| m8gd.24xbesar | 30.0 | 20.0 | 37,5 |
| m8gd.48xbesar | 40,0 | 27.5 | 50,0 |
| m8gd.logam-24xl | 30.0 | 20.0 | 37,5 |
| m8gd.logam-48xl | 40,0 | 27.5 | 50,0 |
| m8i.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| m8i.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| m8i.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| m8i.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| m8i.8xbesar | 10.0 | 6.25 | 12.5 |
| m8i.12xbesar | 15.0 | 9.375 | 18.75 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8i.16xbesar | 20.0 | 12.5 | 25.0 |
| m8i.24xbesar | 30.0 | 20.0 | 37,5 |
| m8i.32xbesar | 40,0 | 27.5 | 50,0 |
| m8i.48xbesar | 60.0 | 41.25 | 75.0 |
| m8i.96xbesar | 80.0 | 55.0 | 100.0 |
| m8i.logam-48xl | 60.0 | 41.25 | 75.0 |
| m8i.logam-96xl | 80.0 | 55.0 | 100.0 |
| m8id.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| m8id.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| m8id.2xbesar | 2.5/10.0 | 1.562/6.25 | 3.125/10.0 |
| m8id.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| m8id.8xbesar | 10.0 | 6.25 | 12.5 |
| m8id.12xbesar | 15.0 | 9.375 | 18.75 |
| m8id.16xbesar | 20.0 | 12.5 | 25.0 |
| m8id.24xbesar | 30.0 | 20.0 | 37,5 |
| m8id.32xbesar | 40,0 | 27.5 | 50,0 |
| m8id.48xbesar | 60.0 | 41.25 | 75.0 |
| m8id.96xbesar | 80.0 | 55.0 | 100.0 |
| m8id.logam-48xl | 60.0 | 41.25 | 75.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m8id.logam-96xl | 80.0 | 55.0 | 100.0 |
| m8i-flex.besar | 0.315/10,0 | 0.198/6.25 | 0.394/10.0 |
| m8i-flex.xlarge | 0,63/10,0 | 0.395/6.25 | 0.788/10.0 |
| m8i-flex.2xbesar | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| m8i-flex.4xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| m8i-flex.8xbesar | 5.0/10.0 | 3.125/6.25 | 6, 25/12, 5 |
| m8i-flex.12xbesar | 7.5/15.0 | 4.687/9.375 | 9.375/18.75 |
| m8i-flex.16xbesar | 10.0/20.0 | 6, 25/12, 5 | 12.5/25.0 |
| m9g.sedang | 0.38/12.0 | 0.242/7.75 | 0.475/12.0 |
| m9g.besar | 0.76/12.0 | 0.51/7.75 | 0.95/ 12.0 |
| m9g.xlarge | 1.5/12.0 | 0.975/7.75 | 1.875/12.0 |
| m9g.2xbesar | 3,0/12,0 | 1.937/7.75 | 3.75/12.0 |
| m9g.4xbesar | 6.0/12.0 | 3.875/7.75 | 7.5/12.0 |
| m9g.8xbesar | 12.0 | 7.75 | 15.0 |
| m9g.12xbesar | 18.0 | 11.75 | 22.5 |
| m9g.16xbesar | 24.0 | 15.5 | 30.0 |
| m9g.24xbesar | 36.0 | 23.5 | 45.0 |
| m9g.48xbesar | 72.0 | 47.0 | 90.0 |
| m9g.logam-48xl | 72.0 | 47.0 | 90.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| m9gd.medium | 0.38/12.0 | 0.242/7.75 | 0.475/12.0 |
| m9gd.besar | 0.76/12.0 | 0.51/7.75 | 0.95/ 12.0 |
| m9gd.xlarge | 1.5/12.0 | 0.975/7.75 | 1.875/12.0 |
| m9gd.2xbesar | 3,0/12,0 | 1.937/7.75 | 3.75/12.0 |
| m9gd.4xbesar | 6.0/12.0 | 3.875/7.75 | 7.5/12.0 |
| m9gd.8xbesar | 12.0 | 7.75 | 15.0 |
| m9gd.12xbesar | 18.0 | 11.75 | 22.5 |
| m9gd.16xbesar | 24.0 | 15.5 | 30.0 |
| m9gd.24xbesar | 36.0 | 23.5 | 45.0 |
| m9gd.48xbesar | 72.0 | 47.0 | 90.0 |
| m9gd.logam-48xl | 72.0 | 47.0 | 90.0 |
| r8a.medium | 0.325/10,0 | 0.195/6.25 | 0.407/10.0 |
| r8a.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| r8a.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| r8a.2xbesar | 2.5/10.0 | 1.562/6.25 | 3.125/10.0 |
| r8a.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| r8a.8xbesar | 10.0 | 6.25 | 12.5 |
| r8a.12xbesar | 15.0 | 9.375 | 18.75 |
| r8a.16xbesar | 20.0 | 12.5 | 25.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r8a.24xbesar | 30.0 | 20.0 | 37,5 |
| r8a.48xbesar | 60.0 | 41.25 | 75.0 |
| r8a.logam-24xl | 30.0 | 20.0 | 37,5 |
| r8a.logam-48xl | 60.0 | 41.25 | 75.0 |
| r8g.sedang | 0.315/10.0 | 0.185/6.25 | 0.394/10.0 |
| r8g.besar | 0,63/10,0 | 0.396/6.25 | 0.788/10.0 |
| r8g.xbesar | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| r8g.2xbesar | 2.5/10.0 | 1.562/6.25 | 3.125/10.0 |
| r8g.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| r8g.8xbesar | 10.0 | 6.25 | 12.5 |
| r8g.12xbesar | 15.0 | 9.375 | 18.75 |
| r8g.16xbesar | 20.0 | 12.5 | 25.0 |
| r8g.24xbesar | 30.0 | 20.0 | 37,5 |
| r8g.48xbesar | 40,0 | 27.5 | 50,0 |
| r8g.logam-24xl | 30.0 | 20.0 | 37,5 |
| r8g.logam-48xl | 40,0 | 27.5 | 50,0 |
| r8gd.medium | 0.315/10,0 | 0.185/6.25 | 0.394/10.0 |
| r8gd.besar | 0,63/10,0 | 0.396/6.25 | 0.788/10.0 |
| r8gd.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r8gd.2xbesar | 2.5/10.0 | 1.562/6.25 | 3.125/10.0 |
| r8gd.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| r8gd.8xbesar | 10.0 | 6.25 | 12.5 |
| r8gd.12xbesar | 15.0 | 9.375 | 18.75 |
| r8gd.16xbesar | 20.0 | 12.5 | 25.0 |
| r8gd.24xbesar | 30.0 | 20.0 | 37,5 |
| r8gd.48xbesar | 40,0 | 27.5 | 50,0 |
| r8gd.logam-24xl | 30.0 | 20.0 | 37,5 |
| r8gd.logam-48xl | 40,0 | 27.5 | 50,0 |
| r8i.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| r8i.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| r8i.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| r8i.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| r8i.8xbesar | 10.0 | 6.25 | 12.5 |
| r8i.12xbesar | 15.0 | 9.375 | 18.75 |
| r8i.16xbesar | 20.0 | 12.5 | 25.0 |
| r8i.24xbesar | 30.0 | 20.0 | 37,5 |
| r8i.32xbesar | 40,0 | 27.5 | 50,0 |
| r8i.48xbesar | 60.0 | 41.25 | 75.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r8i.96xbesar | 80.0 | 55.0 | 100.0 |
| r8i.logam-48xl | 60.0 | 41.25 | 75.0 |
| r8i.logam-96xl | 80.0 | 55.0 | 100.0 |
| r8id.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| r8id.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| r8id.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| r8id.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| r8id.8xbesar | 10.0 | 6.25 | 12.5 |
| r8id.12xbesar | 15.0 | 9.375 | 18.75 |
| r8id.16xbesar | 20.0 | 12.5 | 25.0 |
| r8id.24xbesar | 30.0 | 20.0 | 37,5 |
| r8id.32xbesar | 40,0 | 27.5 | 50,0 |
| r8id.48xbesar | 60.0 | 41.25 | 75.0 |
| r8id.96xbesar | 80.0 | 55.0 | 100.0 |
| r8id.logam-48xl | 60.0 | 41.25 | 75.0 |
| r8id.logam-96xl | 80.0 | 55.0 | 100.0 |
| r8i-flex.besar | 0.315/10.0 | 0.198/6.25 | 0.394/10.0 |
| r8i-flex.xlarge | 0,63/10,0 | 0.395/6.25 | 0.788/10.0 |
| r8i-flex.2xbesar | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r8i-flex.4xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| r8i-flex.8xbesar | 5.0/10.0 | 3.125/6.25 | 6, 25/12, 5 |
| r8i-flex.12xbesar | 7.5/15.0 | 4.687/9.375 | 9.375/18.75 |
| r8i-flex.16xbesar | 10.0/20.0 | 6, 25/12, 5 | 12.5/25.0 |
| r9g.sedang | 0.38/12.0 | 0.242/7.75 | 0.475/12.0 |
| r9g.besar | 0.76/12.0 | 0.51/7.75 | 0.95/ 12.0 |
| r9g.xbesar | 1.5/12.0 | 0.975/7.75 | 1.875/12.0 |
| r9g.2xbesar | 3,0/12,0 | 1.937/7.75 | 3.75/12.0 |
| r9g.4xbesar | 6, 0/12, 0 | 3.875/7.75 | 7.5/12.0 |
| r9g.8xbesar | 12.0 | 7.75 | 15.0 |
| r9g.12xbesar | 18.0 | 11.75 | 22.5 |
| r9g.16xbesar | 24.0 | 15.5 | 30.0 |
| r9g.24xbesar | 36.0 | 23.5 | 45.0 |
| r9g.48xbesar | 72.0 | 47.0 | 90.0 |
| r9g.logam-48xl | 72.0 | 47.0 | 90.0 |
| r9gd.medium | 0, 38/12, 0 | 0.242/7.75 | 0.475/12.0 |
| r9gd.besar | 0.76/12.0 | 0.51/7.75 | 0.95/ 12.0 |
| r9gd.xlarge | 1.5/12.0 | 0.975/7.75 | 1.875/12.0 |
| r9gd.2xbesar | 3,0/12,0 | 1.937/7.75 | 3.75/12.0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| r9gd.4xbesar | 6.0/12.0 | 3.875/7.75 | 7.5/12.0 |
| r9gd.8xbesar | 12.0 | 7.75 | 15.0 |
| r9gd.12xbesar | 18.0 | 11.75 | 22.5 |
| r9gd.16xbesar | 24.0 | 15.5 | 30.0 |
| r9gd.24xbesar | 36.0 | 23.5 | 45.0 |
| r9gd.48xbesar | 72.0 | 47.0 | 90.0 |
| r9gd.logam-48xl | 72.0 | 47.0 | 90.0 |
| x8g.sedang | 0.315/10.0 | 0.185/6.25 | 0.394/10.0 |
| x8g.besar | 0,63/10,0 | 0.396/6.25 | 0.788/10.0 |
| x8g.xbesar | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| x8g.2xbesar | 2.5/10.0 | 1.562/6.25 | 3.125/10.0 |
| x8g.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| x8g.8xbesar | 10.0 | 6.25 | 12.5 |
| x8g.12xbesar | 15.0 | 9.375 | 18.75 |
| x8g.16xbesar | 20.0 | 12.5 | 25.0 |
| x8g.24xbesar | 30.0 | 20.0 | 37,5 |
| x8g.48xbesar | 40,0 | 27.5 | 50,0 |
| x8g.logam-24xl | 30.0 | 20.0 | 37,5 |
| x8g.logam-48xl | 40,0 | 27.5 | 50,0 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| x8aedz.large | 1,25/15,0 | 0.859/10.312 | 1.563/15.0 |
| x8aedz.xlarge | 2.5/15.0 | 1.718/10.312 | 3.125/15.0 |
| x8aedz.3xbesar | 7.5/15.0 | 5.156/10.312 | 9.375/15.0 |
| x8aedz.6xbesar | 15.0 | 10.312 | 18.75 |
| x8aedz.12xbesar | 30.0 | 20.625 | 37,5 |
| x8aedz.24xbesar | 60.0 | 41.25 | 75.0 |
| x8aedz.logam-12xl | 30.0 | 20.625 | 37,5 |
| x8aedz.logam-24xl | 60.0 | 41.25 | 75.0 |
| x8i.besar | 0,65/10,0 | 0.415/6.25 | 0,813/10,0 |
| x8i.xlarge | 1,25/10,0 | 0.781/6.25 | 1.563/10.0 |
| x8i.2xbesar | 2,5/10,0 | 1.562/6.25 | 3.125/10.0 |
| x8i.4xbesar | 5.0/10.0 | 3.125/6.25 | 6,25/10,0 |
| x8i.8xbesar | 10.0 | 6.25 | 12.5 |
| x8i.12xbesar | 15.0 | 9.375 | 18.75 |
| x8i.16xbesar | 20.0 | 12.5 | 25.0 |
| x8i.24xbesar | 30.0 | 20.0 | 37,5 |
| x8i.32xbesar | 40,0 | 27.5 | 50,0 |
| x8i.48xbesar | 60.0 | 41.25 | 75.0 |
| x8i.64xbesar | 70.0 | 50,0 | 87.5 |

| Tipe instans | **default** (Dasar/Burst) | **vpc-1** (Dasar/Burst) | **ebs-1** (Dasar/Burst) |
|---|---|---|---|
| x8i.96xbesar | 80.0 | 55.0 | 100.0 |
| x8i.logam-48xl | 60.0 | 41.25 | 75.0 |
| x8i.logam-96xl | 80.0 | 55.0 | 100.0 |

## Memantau bandwidth instans

Anda dapat menggunakan CloudWatch metrik untuk memantau bandwidth jaringan instance dan paket yang dikirim dan diterima. Anda dapat menggunakan metrik performa jaringan yang disediakan oleh driver Adaptor Jaringan Elastis (ENA) untuk memantau saat lalu lintas melebihi perizinan jaringan yang Amazon EC2 tentukan pada tingkat instans.

Anda dapat mengonfigurasi apakah Amazon EC2 mengirimkan data metrik untuk instans CloudWatch menggunakan periode satu menit atau periode lima menit. Ada kemungkinan bahwa metrik kinerja jaringan akan menunjukkan bahwa tunjangan terlampaui dan paket dijatuhkan sementara metrik CloudWatch instance tidak. Ini dapat terjadi ketika instance memiliki lonjakan singkat dalam permintaan untuk sumber daya jaringan (dikenal sebagai microburst), tetapi CloudWatch metriknya tidak cukup granular untuk mencerminkan lonjakan mikrodetik ini.

Pelajari selengkapnya

- Metrik instans
- Memantau kinerja jaringan

# Jaringan yang disempurnakan di instans Amazon EC2

Jaringan yang disempurnakan menggunakan I/O virtualisasi root tunggal (SR-IOV) untuk menyediakan kemampuan jaringan berkinerja tinggi pada jenis instans yang didukung. SR-IOV adalah metode virtualisasi perangkat yang memberikan I/O kinerja yang lebih tinggi dan pemanfaatan CPU yang lebih rendah jika dibandingkan dengan antarmuka jaringan virtual tradisional. Jaringan yang disempurnakan memberikan bandwidth yang lebih tinggi, kinerja paket per detik (PPS)

yang lebih tinggi, dan latensi yang lebih rendah secara konsisten antar instance. Tidak ada biaya tambahan karena menggunakan jaringan yang ditingkatkan.

Untuk informasi tentang kecepatan jaringan yang didukung untuk setiap tipe instans, lihat Tipe Instans Amazon EC2.

Anda dapat mengaktifkan jaringan yang ditingkatkan menggunakan salah satu dari mekanisme berikut:

Adaptor Jaringan Elastis (ENA)

Adaptor Jaringan Elastis (ENA) mendukung kecepatan jaringan hingga 100 Gbps untuk tipe instans yang didukung.

Semua Nitro-based contoh menggunakan ENA untuk meningkatkan jaringan. Selain itu, Xen-based contoh berikut menggunakan ENA: H1, I3, G3,, P3, P3dn`m4.16xlarge`, dan R4.

Untuk informasi selengkapnya, lihat Aktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda.

Antarmuka Virtual Function (VF) Intel 82599

Antarmuka Virtual Function Intel 82599 mendukung kecepatan jaringan hingga 10 Gbps untuk tipe instans yang didukung.

Tipe instans berikut menggunakan antarmuka Intel 82599 VF untuk jaringan yang ditingkatkan: C3, C4, D2, I2, M4 (tidak termasuk m4.16xlarge), dan R3.

Untuk informasi selengkapnya, lihat Jaringan yang ditingkatkan dengan antarmuka Intel 82599 VF.

Konten


- Aktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda
- Meningkatkan kinerja jaringan antara instans EC2 dengan ENA Express
- Jaringan yang ditingkatkan dengan antarmuka Intel 82599 VF
- Memantau kinerja jaringan untuk pengaturan ENA pada instans EC2 Anda
- Meningkatkan latensi jaringan untuk instans EC2 berbasis Linux
- Pertimbangan sistem nitro untuk penyetelan kinerja
- Mengoptimalkan kinerja jaringan pada instans EC2 Windows

# Aktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda

Amazon EC2 memberikan kemampuan jaringan yang ditingkatkan melalui Adaptor Jaringan Elastis (ENA). Untuk menggunakan jaringan yang disempurnakan, Anda harus menggunakan AMI yang menyertakan driver ENA yang diperlukan atau menginstalnya secara manual. Kemudian Anda dapat mengaktifkan dukungan ENA pada instans Anda.

Untuk meninjau catatan rilis atau petunjuk penginstalan untuk driver ENA, lihat tab yang cocok dengan platform sistem operasi instans Anda.

Linux

Anda dapat meninjau dokumentasi berikut di GitHub:

- Tinjau catatan rilis driver kernel ENA Linux di GitHub.
- Untuk ikhtisar driver kernel ENA Linux yang menyertakan instruksi penginstalan, lihat driver kernel Linux untuk keluarga Elastic Network Adapter (ENA) on GitHub.

Windows

Anda dapat meninjau dokumentasi berikut dari bagian Kelola driver perangkat pada panduan ini:

- Lacak rilis versi driver ENA Windows.
- Instal driver ENA pada instans EC2 Windows.

Sebagai Nitro-based contoh, kemampuan jaringan yang disempurnakan bervariasi menurut versi Nitro yang diimplementasikan oleh jenis instans.

Untuk meninjau spesifikasi jaringan untuk instance Anda, pilih link keluarga instans untuk jenis instans Anda. Jika Anda tidak yakin keluarga instans mana yang berlaku, lihat Konvensi penamaan di panduan Jenis Instans Amazon EC2.

- Spesifikasi jaringan untuk instans komputasi yang dipercepat
- Spesifikasi jaringan untuk instans komputasi yang dioptimalkan
- Spesifikasi jaringan untuk contoh tujuan umum
- Spesifikasi jaringan untuk instans komputasi berkinerja tinggi
- Spesifikasi jaringan untuk instans yang dioptimalkan memori
- Spesifikasi jaringan untuk instans yang dioptimalkan penyimpanan

Daftar Isi


- Prasyarat untuk peningkatan jaringan dengan ENA
- Menguji apakah jaringan yang ditingkatkan diaktifkan
- Mengaktifkan jaringan yang ditingkatkan pada instans Anda
- Antrian ENA
- Memecahkan masalah driver kernel ENA di Linux
- Memecahkan masalah driver Windows Adaptor Jaringan Elastic


## Prasyarat untuk peningkatan jaringan dengan ENA

Untuk mempersiapkan jaringan yang ditingkatkan menggunakan ENA, siapkan instans Anda sebagai berikut:

- Luncurkan Nitro-based instance.
- Pastikan instans tersebut memiliki konektivitas internet.
- Jika memiliki data penting pada instans yang ingin Anda pertahankan, Anda harus melakukan back up data tersebut sekarang dengan membuat AMI dari instans Anda. Memperbarui driver kernel ENA dan mengaktifkan `enaSupport` atribut dapat membuat instance atau sistem operasi yang tidak kompatibel tidak dapat dijangkau. Jika Anda memiliki back up terbaru, data Anda akan tetap disimpan jika hal ini terjadi.
- Instans Linux — Luncurkan instance menggunakan versi kernel Linux yang didukung dan distribusi yang didukung, sehingga jaringan yang ditingkatkan ENA diaktifkan untuk instans Anda secara otomatis. Untuk informasi selengkapnya, lihat Catatan Rilis ENA Linux Kernel Driver.
- Instans Windows — Jika instance menjalankan Windows Server 2008 R2 SP1, pastikan instans tersebut memiliki pembaruan dukungan penandatanganan SHA-2 kode.
- Gunakan AWS CloudShell dari Konsol Manajemen AWS, atau instal dan konfigurasikan AWS CLI atau AWS Tools for Windows PowerShell pada komputer mana pun yang Anda pilih, sebaiknya desktop atau laptop lokal Anda. Untuk informasi selengkapnya, lihat Akses Amazon EC2 atau Panduan Pengguna AWS CloudShell. Jaringan yang ditingkatkan tidak dapat dikelola dari konsol Amazon EC2.

## Menguji apakah jaringan yang ditingkatkan diaktifkan

Anda dapat menguji apakah jaringan yang disempurnakan diaktifkan di instans atau AMI Anda.

Atribut instance

Periksa nilai atribut `enaSupport` instance.

AWS CLI

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query "Reservations[].Instances[].EnaSupport"
```

Jika jaringan yang disempurnakan diaktifkan, outputnya adalah sebagai berikut.

```
[
    true
]
```

PowerShell

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances.EnaSupport
```

Jika jaringan yang disempurnakan diaktifkan, outputnya adalah sebagai berikut.

```
True
```

Atribut gambar

Periksa nilai atribut `enaSupport` gambar.

AWS CLI

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-id ami-0abcdef1234567890 \
    --query "Images[].EnaSupport"
```

Jika jaringan yang disempurnakan diaktifkan, outputnya adalah sebagai berikut.

```
[
    true
]
```

PowerShell

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).EnaSupport
```

Jika jaringan yang disempurnakan diaktifkan, outputnya adalah sebagai berikut.

```
True
```

Driver antarmuka jaringan Linux

Gunakan perintah berikut untuk memverifikasi bahwa driver `ena` kernel sedang digunakan pada antarmuka tertentu, menggantikan nama antarmuka yang ingin Anda periksa. Jika Anda menggunakan antarmuka tunggal (default), ini adalah `eth0`. Jika distribusi Linux Anda mendukung nama jaringan yang dapat diprediksi, ini bisa menjadi nama seperti`ens5`. Untuk informasi selengkapnya, perluas bagian untuk RHEL, SUSE, dan CentOS di. Mengaktifkan jaringan yang ditingkatkan pada instans Anda

Dalam contoh berikut, driver `ena` kernel tidak dimuat, karena driver yang terdaftar adalah`vif`.

```
[ec2-user ~]$ ethtool -i eth0
driver: vif
version:
firmware-version:
bus-info: vif-0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

Dalam contoh ini, driver `ena` kernel dimuat dan pada versi minimum yang disarankan. Instans ini memiliki jaringan yang ditingkatkan, yang dikonfigurasi dengan benar.

```
[ec2-user ~]$ ethtool -i eth0
driver: ena
version: 1.5.0g
firmware-version:
expansion-rom-version:
bus-info: 0000:00:05.0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

## Mengaktifkan jaringan yang ditingkatkan pada instans Anda

Prosedur yang Anda gunakan tergantung pada sistem operasi instance.

### Amazon Linux

AMI untuk Amazon Linux menyertakan driver kernel yang diperlukan untuk jaringan yang ditingkatkan dengan ENA diinstal dan dukungan ENA diaktifkan. Oleh karena itu, jika Anda meluncurkan instans dengan versi HVM dari Amazon Linux pada tipe instans yang didukung, jaringan yang ditingkatkan telah diaktifkan untuk instans Anda. Untuk informasi selengkapnya, lihat Menguji apakah jaringan yang ditingkatkan diaktifkan.

### Ubuntu

AMI Ubuntu HVM terbaru menyertakan driver kernel yang diperlukan untuk jaringan yang ditingkatkan dengan ENA diinstal dan dukungan ENA diaktifkan. Oleh karena itu, jika Anda meluncurkan instans dengan AMI HVM Ubuntu terbaru pada tipe instans yang didukung, jaringan yang ditingkatkan sudah diaktifkan untuk instans Anda. Untuk informasi selengkapnya, lihat Menguji apakah jaringan yang ditingkatkan diaktifkan.

Jika Anda meluncurkan instans menggunakan AMI lama dan jaringan yang ditingkatkan belum diaktifkan, Anda dapat menginstal paket kernel `linux-aws` untuk mendapatkan driver jaringan terbaru yang ditingkatkan dan mengupdate atribut yang diperlukan.

Untuk menginstal paket **`kernel`** linux-aws (Ubuntu 16.04 atau yang lebih baru)

Ubuntu 16.04 dan 18.04 dikirimkan dengan kernel kustom Ubuntu (paket kernel `linux-aws`). Untuk menggunakan kernel yang berbeda, hubungi Dukungan.

Untuk menginstal paket **`kernel`** linux-aws (Ubuntu Trusty 14.04)

1. Hubungkan ke instans Anda.
2. Perbarui cache paket dan paket.

```
ubuntu:~$ sudo apt-get update && sudo apt-get upgrade -y linux-aws
```

> **Important**
>
> Jika selama proses pembaruan Anda diminta untuk menginstal `grub`, gunakan `/dev/xvda` untuk menginstal `grub`, lalu pilih untuk mempertahankan versi `/boot/grub/menu.lst` saat ini.

3. [EBS-backed instance] Dari komputer lokal Anda, hentikan instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: stop-instances (AWS CLI) atau Stop-EC2Instance (AWS Tools for Windows PowerShell).

   [Instans yang didukung penyimpanan instans] Anda tidak dapat menghentikan instans untuk memodifikasi atribut. Sebagai gantinya, lanjutkan ke prosedur ini: Untuk mengaktifkan jaringan yang ditingkatkan di Ubuntu (instans yang didukung penyimpanan instans).
4. Dari komputer lokal Anda, aktifkan atribut jaringan yang ditingkatkan menggunakan salah satu dari perintah berikut ini:

   - modify-instance-attribute (AWS CLI)

     ```
     aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --ena-support
     ```

   - Edit-EC2InstanceAttribute(Alat untuk Windows PowerShell)

     ```
     Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -EnaSupport $true
     ```
5. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di Buat EBS-backed AMI Amazon. AMI mewarisi atribut `enaSupport` jaringan yang ditingkatkan dari instans. Oleh karena itu, Anda dapat menggunakan AMI ini untuk meluncurkan instans lain dengan jaringan yang ditingkatkan diaktifkan secara default.
6. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: start-instances (AWS CLI) atau Start-EC2Instance (AWS Tools for Windows PowerShell).

Untuk mengaktifkan jaringan yang ditingkatkan di Ubuntu (instans yang didukung penyimpanan instans)

Ikuti prosedur sebelumnya hingga langkah tempat Anda menghentikan instans. Buat AMI baru seperti yang dijelaskan di Buat S3-backed AMI Amazon, pastikan untuk mengaktifkan atribut jaringan yang ditingkatkan saat Anda mendaftarkan AMI.

- register-image (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- Register-EC2Image (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport $true ...
```

RHEL, SUSE, CentOS

AMI terbaru untuk Red Hat Enterprise Linux, SUSE Linux Enterprise Server, dan CentOS menyertakan driver kernel yang diperlukan untuk jaringan yang ditingkatkan dengan ENA dan mendukung ENA diaktifkan. Oleh karena itu, jika Anda meluncurkan instans dengan AMI terbaru pada tipe instans yang didukung, jaringan yang ditingkatkan telah diaktifkan untuk instans Anda. Untuk informasi selengkapnya, lihat Menguji apakah jaringan yang ditingkatkan diaktifkan.

Prosedur berikut menyediakan langkah-langkah umum untuk mengaktifkan jaringan yang ditingkatkan pada distribusi Linux selain Amazon Linux AMI atau Ubuntu. Untuk informasi selengkapnya, seperti sintaksis terperinci untuk perintah, lokasi file, atau paket dan dukungan alat, lihat dokumentasi untuk distribusi Linux Anda.

Untuk mengaktifkan jaringan yang ditingkatkan di Linux

1. Terhubung ke instans Anda.
2. Kloning kode sumber untuk driver `ena` kernel pada instance Anda dari GitHub athttps://github.com/amzn/amzn-drivers. (SUSE Linux Enterprise Server 12 SP2 dan yang lebih baru menyertakan ENA 2.02 secara default, jadi Anda tidak diharuskan untuk mengunduh dan mengompilasi driver ENA. Untuk SUSE Linux Enterprise Server 12 SP2 dan yang lebih baru, Anda harus mengajukan permintaan untuk menambahkan versi driver yang Anda inginkan ke kernel stok).

```
git clone https://github.com/amzn/amzn-drivers
```

3. Kompilasi dan instal driver `ena` kernel pada instance Anda. Langkah-langkah ini bergantung pada distribusi Linux. Untuk informasi selengkapnya tentang mengkompilasi driver kernel di Red Hat Enterprise Linux, lihat Bagaimana cara menginstal driver ENS terbaru untuk dukungan jaringan yang ditingkatkan pada instans Amazon EC2 yang menjalankan RHEL?
4. Jalankan sudo depmod perintah untuk memperbarui dependensi driver kernel.
5. Perbarui `initramfs` pada instance Anda untuk memastikan bahwa driver kernel baru dimuat pada saat boot. Misalnya, jika distribusi Anda mendukung dracut, Anda dapat menggunakan perintah berikut.

```
dracut -f -v
```

6. Tentukan apakah sistem Anda menggunakan nama antarmuka jaringan yang dapat diprediksi secara default. Sistem yang menggunakan systemd atau udev versi 197 atau lebih tinggi dapat mengganti nama perangkat Ethernet dan tidak menjamin bahwa satu antarmuka jaringan akan dinamai `eth0`. Perilaku ini dapat menyebabkan masalah saat terhubung ke instans Anda. Untuk informasi lebih lanjut dan untuk melihat opsi konfigurasi lainnya, lihat Nama Antarmuka Jaringan yang Dapat Diprediksi di situs web freedesktop.org.

   a. Anda dapat memeriksa udev versi systemd atau pada RPM-based sistem dengan perintah berikut.

   ```
   rpm -qa | grep -e '^systemd-[0-9]\+\|^udev-[0-9]\+'
   systemd-208-11.el7_0.2.x86_64
   ```

   Dalam contoh Red Hat Enterprise Linux 7 ini, ver systemd sinya adalah 208, jadi nama antarmuka jaringan yang dapat diprediksi harus dinonaktifkan.

   b. Nonaktifkan nama antarmuka jaringan yang dapat diprediksi dengan menambahkan opsi `net.ifnames=0` ke baris `GRUB_CMDLINE_LINUX` di `/etc/default/grub`.

   ```
   sudo sed -i '/^GRUB\_CMDLINE\_LINUX/s/\"$/\ net\.ifnames\=0\"/' /etc/default/
   grub
   ```

   c. Buat ulang file konfigurasi grub.

   ```
   sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

7. [EBS-backed instance] Dari komputer lokal Anda, hentikan instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: stop-instances (AWS CLI), Stop-EC2Instance (AWS Tools for Windows PowerShell).

   [Instans yang didukung penyimpanan instans] Anda tidak dapat menghentikan instans untuk memodifikasi atribut. Sebagai gantinya, lanjutkan ke prosedur ini: Untuk mengaktifkan jaringan yang ditingkatkan di Linux (instans yang didukung penyimpanan instans).

8. Dari komputer lokal Anda, aktifkan atribut `enaSupport` jaringan yang ditingkatkan menggunakan salah satu dari perintah berikut ini:

   - modify-instance-attribute (AWS CLI)

     ```
     aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --ena-support
     ```

   - Edit-EC2InstanceAttribute(Alat untuk Windows PowerShell)

     ```
     Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -EnaSupport $true
     ```

9. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di Buat EBS-backed AMI Amazon. AMI mewarisi atribut `enaSupport` jaringan yang ditingkatkan dari instans. Oleh karena itu, Anda dapat menggunakan AMI ini untuk meluncurkan instans lain dengan jaringan yang ditingkatkan diaktifkan secara default.

   Jika sistem operasi instans Anda berisi file `/etc/udev/rules.d/70-persistent-net.rules`, Anda harus menghapusnya sebelum membuat AMI. File ini berisi alamat MAC untuk adaptor Ethernet dari instans asli. Jika instans lain melakukan booting dengan file ini, sistem operasi tersebut `eth0` tidak akan dapat menemukan perangkat dan mungkin gagal, yang menyebabkan masalah booting. File ini dibuat ulang pada siklus boot berikutnya, dan setiap instans yang diluncurkan dari AMI membuat versi file mereka sendiri.

10. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: start-instances (AWS CLI) atau Start-EC2Instance (AWS Tools for Windows PowerShell).

11. (Opsional) Sambungkan ke instance Anda dan verifikasi bahwa driver kernel diinstal.

    Jika Anda tidak dapat terhubung ke instans Anda setelah mengaktifkan jaringan yang ditingkatkan, lihat Memecahkan masalah driver kernel ENA di Linux.

Untuk mengaktifkan jaringan yang ditingkatkan di Linux (instans yang didukung penyimpanan instans)

Ikuti prosedur sebelumnya hingga langkah tempat Anda menghentikan instans. Buat AMI baru seperti yang dijelaskan di Buat S3-backed AMI Amazon, pastikan untuk mengaktifkan atribut jaringan yang ditingkatkan saat Anda mendaftarkan AMI.

- register-image (AWS CLI)

```
aws ec2 register-image --ena-support ...
```

- Register-EC2Image (AWS Tools for Windows PowerShell)

```
Register-EC2Image -EnaSupport ...
```

Ubuntu dengan DKMS

Metode ini hanya untuk tujuan pengujian dan umpan balik. Metode ini tidak dimaksudkan untuk digunakan dengan penerapan produksi. Untuk penerapan produksi, lihat Ubuntu.

> **Important**
>
> Menggunakan DKMS membatalkan perjanjian dukungan untuk langganan Anda. Ini tidak boleh digunakan untuk penerapan produksi.

Untuk mengaktifkan jaringan yang ditingkatkan dengan ENA di Ubuntu (EBS-backed instance)

1. Ikuti langkah 1 dan 2 dalam Ubuntu.
2. Instal `build-essential` paket untuk mengkompilasi driver kernel dan `dkms` paket sehingga driver `ena` kernel Anda dibangun kembali setiap kali kernel Anda diperbarui.

   ```
   ubuntu:~$ sudo apt-get install -y build-essential dkms
   ```

3. Kloning sumber untuk driver `ena` kernel pada instance Anda dari GitHub athttps://github.com/amzn/amzn-drivers.

   ```
   ubuntu:~$ git clone https://github.com/amzn/amzn-drivers
   ```

4. Pindahkan paket `amzn-drivers` ke direktori `/usr/src/` sehingga DKMS dapat menemukannya dan membangunnya untuk setiap pembaruan kernel. Tambahkan nomor versi (Anda dapat menemukan nomor versi saat ini di catatan rilis) dari kode sumber ke nama direktori. Misalnya, versi `1.0.0` ditunjukkan pada contoh berikut.

```
ubuntu:~$ sudo mv amzn-drivers /usr/src/amzn-drivers-1.0.0
```

5. Buat file konfigurasi DKMS menggunakan nilai berikut, dengan menggantikan versi `ena`.

   Buat mengajukan.

```
ubuntu:~$ sudo touch /usr/src/amzn-drivers-1.0.0/dkms.conf
```

   Edit file dan tambahkan nilai berikut.

```
ubuntu:~$ sudo vim /usr/src/amzn-drivers-1.0.0/dkms.conf
PACKAGE_NAME="ena"
PACKAGE_VERSION="1.0.0"
CLEAN="make -C kernel/linux/ena clean"
MAKE="make -C kernel/linux/ena/ BUILD_KERNEL=${kernelver}"
BUILT_MODULE_NAME[0]="ena"
BUILT_MODULE_LOCATION="kernel/linux/ena"
DEST_MODULE_LOCATION[0]="/updates"
DEST_MODULE_NAME[0]="ena"
AUTOINSTALL="yes"
```

6. Tambahkan, buat, dan instal driver `ena` kernel pada instance Anda menggunakan DKMS.

   Tambahkan driver kernel ke DKMS.

```
ubuntu:~$ sudo dkms add -m amzn-drivers -v 1.0.0
```

   Bangun driver kernel menggunakan dkms perintah.

```
ubuntu:~$ sudo dkms build -m amzn-drivers -v 1.0.0
```

   Instal driver kernel menggunakandkms.

```
ubuntu:~$ sudo dkms install -m amzn-drivers -v 1.0.0
```

7. Bang `initramfs` un ulang sehingga driver kernel yang benar dimuat pada saat boot.

```
ubuntu:~$ sudo update-initramfs -u -k all
```

8. Verifikasi bahwa driver `ena` kernel diinstal menggunakan perintah modinfo ena dariMenguji apakah jaringan yang ditingkatkan diaktifkan.

```
ubuntu:~$ modinfo ena
filename:    /lib/modules/3.13.0-74-generic/updates/dkms/ena.ko
version:  1.0.0
license:  GPL
description: Elastic Network Adapter (ENA)
author:   Amazon.com, Inc. or its affiliates
srcversion:  9693C876C54CA64AE48F0CA
alias:    pci:v00001D0Fd0000EC21sv*sd*bc*sc*i*
alias:    pci:v00001D0Fd0000EC20sv*sd*bc*sc*i*
alias:    pci:v00001D0Fd00001EC2sv*sd*bc*sc*i*
alias:    pci:v00001D0Fd00000EC2sv*sd*bc*sc*i*
depends:
vermagic:    3.13.0-74-generic SMP mod_unload modversions
parm:     debug:Debug level (0=none,...,16=all) (int)
parm:     push_mode:Descriptor / header push mode (0=automatic,1=disable,3=enable)
    0 - Automatically choose according to device capability (default)
    1 - Don't push anything to device memory
    3 - Push descriptors and header buffer to device memory (int)
parm:     enable_wd:Enable keepalive watchdog (0=disable,1=enable,default=1) (int)
parm:     enable_missing_tx_detection:Enable missing Tx completions. (default=1)
 (int)
parm:     numa_node_override_array:Numa node override map
(array of int)
parm:     numa_node_override:Enable/Disable numa node override (0=disable)
(int)
```

9. Lanjutkan dengan Langkah 3 di Ubuntu.

### Mengaktifkan jaringan yang ditingkatkan di Windows

Jika Anda meluncurkan instans dan instans tersebut belum mengaktifkan jaringan yang ditingkatkan, Anda harus mengunduh dan menginstal driver adaptor jaringan yang diperlukan pada instans Anda, lalu menyetel atribut instans `enaSupport` untuk mengaktifkan jaringan yang ditingkatkan.

Untuk mengaktifkan jaringan yang ditingkatkan

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
2. [Hanya Windows Server 2016 dan 2019] Jalankan PowerShell skrip EC2launch berikut untuk mengonfigurasi instance setelah driver diinstal.

```
PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -
Schedule
```

3. Dari instans, instal driver sebagai berikut:
   a. Unduh driver terbaru ke instans.
   b. Ekstrak arsip zip.
   c. Instal driver dengan menjalankan `install.ps1` PowerShell skrip.

   > **Note**
   >
   > Jika Anda mendapatkan kesalahan kebijakan eksekusi, atur kebijakan ke `Unrestricted` (secara default kebijakan ini diatur ke `Restricted` atau `RemoteSigned`). Di baris perintah, jalankan`Set-ExecutionPolicy -ExecutionPolicy Unrestricted`, lalu jalankan `install.ps1` PowerShell skrip lagi.

4. Dari komputer lokal Anda, hentikan instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: stop-instances (AWS CLI) atau Stop-EC2Instance (AWS Tools for Windows PowerShell).
5. Aktifkan dukungan ENA pada instans Anda sebagai berikut:
   a. Dari komputer lokal Anda, periksa atribut dukungan ENA instans EC2 pada instans Anda dengan menjalankan salah satu perintah berikut. Jika atribut tersebut tidak diaktifkan, output akan menjadi "[]" atau kosong. `EnaSupport` diatur ke `false` secara default.
      - describe-instances (AWS CLI)

        ```
        aws ec2 describe-instances --instance-ids i-1234567890abcdef0 --query
         "Reservations[].Instances[].EnaSupport"
        ```

      - Get-EC2Instance(Alat untuk Windows PowerShell)

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances.EnaSupport
```

b. Untuk mengaktifkan dukungan ENA, jalankan salah satu dari perintah berikut ini:

- modify-instance-attribute (AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --ena-
support
```

- Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -EnaSupport $true
```

Jika Anda mengalami masalah saat memulai ulang instans, Anda juga dapat menonaktifkan dukungan ENA menggunakan salah satu dari perintah berikut:

- modify-instance-attribute (AWS CLI)

```
aws ec2 modify-instance-attribute --instance-id i-1234567890abcdef0 --no-ena-
support
```

- Edit-EC2InstanceAttribute (AWS Tools for Windows PowerShell)

```
Edit-EC2InstanceAttribute -InstanceId i-1234567890abcdef0 -EnaSupport $false
```

c. Verifikasi bahwa atribut telah diatur ke `true` menggunakan describe-instances atau Get-EC2Instance seperti yang ditunjukkan sebelumnya. Anda seharusnya sekarang melihat output berikut:

```
[
 true
]
```

6. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: start-instances (AWS CLI) atau Start-EC2Instance (AWS Tools for Windows PowerShell).
7. Pada instans ini, validasi bahwa driver ENA diinstal dan diaktifkan sebagai berikut:

a. Buka menu konteks (klik kanan) untuk ikon jaringan dan pilih Buka Jaringan dan Pusat Berbagi.

b. Pilih adaptor Ethernet (misalnya, Ethernet 2).

c. Pilih Detail. Untuk Detail Koneksi Jaringan, periksa apakah Deskripsi adalah Amazon Elastic Network Adapter.

8. (Opsional) Buat AMI dari instans. AMI mewarisi atribut `enaSupport` dari instans tersebut. Oleh karena itu, Anda dapat menggunakan AMI ini untuk meluncurkan instans lain dengan ENA yang diaktifkan secara default.

## Antrian ENA

Antrian Elastic Network Adapter (ENA) menangani pemrosesan paket untuk antarmuka jaringan Amazon EC2 Anda. Setiap jenis instans memiliki sejumlah antrean ENA yang dibagikan di seluruh Elastic Network Interfaces (ENI) nya. Secara default, setiap ENI mendapatkan jumlah antrian tetap berdasarkan jenis dan ukuran instance.

Dengan alokasi antrian ENA yang dapat dikonfigurasi, Anda dapat memilih berapa banyak antrian yang didapat setiap ENI. Total per ENI dan per instans masih dibatasi. Ini membantu Anda mencocokkan pemrosesan paket dengan beban kerja Anda. Misalnya, Anda dapat memberikan lebih banyak antrian ke ENI dengan lalu lintas padat dan lebih sedikit ke satu dengan lalu lintas ringan.

Topik


- Spesifikasi antrian ENA untuk tipe instans EC2
- Konfigurasikan alokasi antrian ENA


### Spesifikasi antrian ENA untuk tipe instans EC2

Tabel berikut mencantumkan jenis instance yang mendukung alokasi antrian ENA yang dapat dikonfigurasi, bersama dengan nilai antrian default dan maksimumnya.

> **Note**
>
> Untuk nilai antrian ENA untuk semua jenis instans, termasuk jenis yang tidak mendukung alokasi antrian ENA yang dapat dikonfigurasi, lihat tabel spesifikasi jaringan di panduan Jenis Instans Amazon EC2.

General purpose

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m6i.large | 2 | 2 | 6 |
| m6i.xlarge | 4 | 4 | 16 |
| m6i.2xlarge | 8 | 8 | 32 |
| m6i.4xlarge | 8 | 16 | 64 |
| m6i.8xlarge | 8 | 32 | 64 |
| m6i.12xlarge | 8 | 32 | 64 |
| m6i.16xlarge | 8 | 32 | 120 |
| m6i.24xlarge | 8 | 32 | 120 |
| m6i.32xlarge | 8 | 32 | 120 |
| m6id.large | 2 | 2 | 6 |
| m6id.xlarge | 4 | 4 | 16 |
| m6id.2xlarge | 8 | 8 | 32 |
| m6id.4xlarge | 8 | 16 | 64 |
| m6id.8xlarge | 8 | 32 | 64 |
| m6id.12xlarge | 8 | 32 | 64 |
| m6id.16xlarge | 8 | 32 | 120 |
| m6id.24xlarge | 8 | 32 | 120 |
| m6id.32xlarge | 8 | 32 | 120 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m6idn.large | 2 | 2 | 6 |
| m6idn.xlarge | 4 | 4 | 16 |
| m6idn.2xlarge | 8 | 8 | 32 |
| m6idn.4xlarge | 8 | 16 | 64 |
| m6idn.8xlarge | 16 | 32 | 128 |
| m6idn.12xlarge | 16 | 32 | 128 |
| m6idn.16xlarge | 16 | 32 | 240 |
| m6idn.24xlarge | 32 | 32 | 480 |
| m6idn.32xlarge | 32 | 32 | 512* |
| m6in.large | 2 | 2 | 6 |
| m6in.xlarge | 4 | 4 | 16 |
| m6in.2xlarge | 8 | 8 | 32 |
| m6in.4xlarge | 8 | 16 | 64 |
| m6in.8xlarge | 16 | 32 | 128 |
| m6in.12xlarge | 16 | 32 | 128 |
| m6in.16xlarge | 16 | 32 | 240 |
| m6in.24xlarge | 32 | 32 | 480 |
| m6in.32xlarge | 32 | 32 | 512* |
| m8a.medium | 1 | 1 | 3 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8a.large | 2 | 2 | 6 |
| m8a.xlarge | 4 | 4 | 16 |
| m8a.2xlarge | 8 | 8 | 32 |
| m8a.4xlarge | 8 | 16 | 64 |
| m8a.8xlarge | 8 | 32 | 128 |
| m8a.12xlarge | 16 | 64 | 192 |
| m8a.16xlarge | 16 | 64 | 256 |
| m8a.24xlarge | 16 | 128 | 384 |
| m8a.48xlarge | 32 | 128 | 768 |
| m8a.metal-24xl | 16 | 128 | 384 |
| m8a.metal-48xl | 32 | 128 | 768 |
| m8azn.medium | 1 | 1 | 3 |
| m8azn.large | 2 | 2 | 8 |
| m8azn.xlarge | 4 | 4 | 16 |
| m8azn.3xlarge | 4 | 16 | 48 |
| m8azn.6xlarge | 8 | 32 | 96 |
| m8azn.12xlarge | 8 | 64 | 192 |
| m8azn.24xlarge | 16 | 128 | 384 |
| m8azn.metal-12xl | 8 | 64 | 192 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8azn.metal-24xl | 16 | 128 | 384 |
| m8gb.medium | 1 | 1 | 2 |
| m8gb.large | 2 | 2 | 6 |
| m8gb.xlarge | 4 | 4 | 16 |
| m8gb.2xlarge | 8 | 8 | 32 |
| m8gb.4xlarge | 8 | 16 | 64 |
| m8gb.8xlarge | 8 | 32 | 128 |
| m8gb.12xlarge | 16 | 64 | 192 |
| m8gb.16xlarge | 16 | 64 | 256 |
| m8gb.24xlarge | 16 | 128 | 384 |
| m8gb.48xlarge | 32 | 128 | 768* |
| m8gb.metal-24xl | 32 | 128 | 768 |
| m8gb.metal-48xl | 32 | 128 | 768* |
| m8gn.medium | 1 | 1 | 2 |
| m8gn.large | 2 | 2 | 6 |
| m8gn.xlarge | 4 | 4 | 16 |
| m8gn.2xlarge | 8 | 8 | 32 |
| m8gn.4xlarge | 8 | 16 | 64 |
| m8gn.8xlarge | 8 | 32 | 128 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8gn.12xlarge | 16 | 64 | 192 |
| m8gn.16xlarge | 16 | 64 | 256 |
| m8gn.24xlarge | 16 | 128 | 384 |
| m8gn.48xlarge | 32 | 128 | 768* |
| m8gn.metal-24xl | 32 | 128 | 768 |
| m8gn.metal-48xl | 32 | 128 | 768* |
| m8i.large | 2 | 2 | 6 |
| m8i.xlarge | 4 | 4 | 16 |
| m8i.2xlarge | 8 | 8 | 32 |
| m8i.4xlarge | 8 | 16 | 64 |
| m8i.8xlarge | 8 | 32 | 128 |
| m8i.12xlarge | 16 | 64 | 192 |
| m8i.16xlarge | 16 | 64 | 256 |
| m8i.24xlarge | 16 | 128 | 384 |
| m8i.32xlarge | 16 | 128 | 512 |
| m8i.48xlarge | 32 | 128 | 768 |
| m8i.96xlarge | 32 | 128 | 1536 |
| m8i.metal-48xl | 32 | 128 | 768 |
| m8i.metal-96xl | 32 | 128 | 1536 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8id.large | 2 | 2 | 6 |
| m8id.xlarge | 4 | 4 | 16 |
| m8id.2xlarge | 8 | 8 | 32 |
| m8id.4xlarge | 8 | 16 | 64 |
| m8id.8xlarge | 8 | 32 | 128 |
| m8id.12xlarge | 16 | 64 | 192 |
| m8id.16xlarge | 16 | 64 | 256 |
| m8id.24xlarge | 16 | 128 | 384 |
| m8id.32xlarge | 16 | 128 | 512 |
| m8id.48xlarge | 32 | 128 | 768 |
| m8id.96xlarge | 32 | 128 | 1536 |
| m8id.metal-48xl | 32 | 128 | 768 |
| m8id.metal-96xl | 32 | 128 | 1536 |
| m8i-flex.large | 1 | 1 | 3 |
| m8i-flex.xlarge | 2 | 2 | 8 |
| m8i-flex.2xlarge | 4 | 4 | 16 |
| m8i-flex.4xlarge | 4 | 8 | 32 |
| m8i-flex.8xlarge | 4 | 16 | 64 |
| m8i-flex.12xlarge | 8 | 32 | 96 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8i-flex.16xlarge | 8 | 32 | 128 |
| m8in.large | 2 | 2 | 8 |
| m8in.xlarge | 4 | 4 | 16 |
| m8in.2xlarge | 8 | 8 | 32 |
| m8in.4xlarge | 8 | 16 | 64 |
| m8in.8xlarge | 16 | 32 | 128 |
| m8in.12xlarge | 16 | 64 | 192 |
| m8in.16xlarge | 16 | 64 | 256 |
| m8in.24xlarge | 16 | 128 | 256 |
| m8in.32xlarge | 32 | 128 | 512 |
| m8in.48xlarge | 32 | 128 | 768 |
| m8in.96xlarge | 32 | 128 | 1536* |
| m8in.metal-48xl | 32 | 128 | 768 |
| m8in.metal-96xl | 32 | 128 | 1536* |
| m8idn.large | 2 | 2 | 8 |
| m8idn.xlarge | 4 | 4 | 16 |
| m8idn.2xlarge | 8 | 8 | 32 |
| m8idn.4xlarge | 8 | 16 | 64 |
| m8idn.8xlarge | 16 | 32 | 128 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8idn.12xlarge | 16 | 64 | 192 |
| m8idn.16xlarge | 16 | 64 | 256 |
| m8idn.24xlarge | 16 | 128 | 256 |
| m8idn.32xlarge | 32 | 128 | 512 |
| m8idn.48xlarge | 32 | 128 | 768 |
| m8idn.96xlarge | 32 | 128 | 1536* |
| m8idn.metal-48xl | 32 | 128 | 768 |
| m8idn.metal-96xl | 32 | 128 | 1536* |
| m8ine.large | 2 | 2 | 8 |
| m8ine.xlarge | 4 | 4 | 16 |
| m8ine.2xlarge | 8 | 8 | 32 |
| m8ine.4xlarge | 16 | 16 | 128 |
| m8ine.8xlarge | 32 | 32 | 256 |
| m8ine.12xlarge | 32 | 64 | 384 |
| m8ib.large | 2 | 2 | 8 |
| m8ib.xlarge | 4 | 4 | 16 |
| m8ib.2xlarge | 8 | 8 | 32 |
| m8ib.4xlarge | 8 | 16 | 64 |
| m8ib.8xlarge | 16 | 32 | 128 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8ib.12xlarge | 16 | 64 | 192 |
| m8ib.16xlarge | 16 | 64 | 256 |
| m8ib.24xlarge | 16 | 128 | 256 |
| m8ib.32xlarge | 32 | 128 | 512 |
| m8ib.48xlarge | 32 | 128 | 768 |
| m8ib.96xlarge | 32 | 128 | 1536* |
| m8ib.metal-48xl | 32 | 128 | 768 |
| m8ib.metal-96xl | 32 | 128 | 1536* |
| m8idb.large | 2 | 2 | 8 |
| m8idb.xlarge | 4 | 4 | 16 |
| m8idb.2xlarge | 8 | 8 | 32 |
| m8idb.4xlarge | 8 | 16 | 64 |
| m8idb.8xlarge | 16 | 32 | 128 |
| m8idb.12xlarge | 16 | 64 | 192 |
| m8idb.16xlarge | 16 | 64 | 256 |
| m8idb.24xlarge | 16 | 128 | 256 |
| m8idb.32xlarge | 32 | 128 | 512 |
| m8idb.48xlarge | 32 | 128 | 768 |
| m8idb.96xlarge | 32 | 128 | 1536* |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m8idb.metal-48xl | 32 | 128 | 768 |
| m8idb.metal-96xl | 32 | 128 | 1536* |
| m9g.medium | 1 | 1 | 2 |
| m9g.large | 2 | 2 | 6 |
| m9g.xlarge | 4 | 4 | 16 |
| m9g.2xlarge | 8 | 8 | 32 |
| m9g.4xlarge | 8 | 16 | 64 |
| m9g.8xlarge | 8 | 32 | 128 |
| m9g.12xlarge | 16 | 64 | 192 |
| m9g.16xlarge | 16 | 64 | 256 |
| m9g.24xlarge | 16 | 128 | 384 |
| m9g.48xlarge | 32 | 128 | 768 |
| m9g.metal-48xl | 32 | 128 | 768 |
| m9gd.medium | 1 | 1 | 2 |
| m9gd.large | 2 | 2 | 6 |
| m9gd.xlarge | 4 | 4 | 16 |
| m9gd.2xlarge | 8 | 8 | 32 |
| m9gd.4xlarge | 8 | 16 | 64 |
| m9gd.8xlarge | 8 | 32 | 128 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| m9gd.12xlarge | 16 | 64 | 192 |
| m9gd.16xlarge | 16 | 64 | 256 |
| m9gd.24xlarge | 16 | 128 | 384 |
| m9gd.48xlarge | 32 | 128 | 768 |
| m9gd.metal-48xl | 32 | 128 | 768 |

Compute optimized

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c6i.large | 2 | 2 | 6 |
| c6i.xlarge | 4 | 4 | 16 |
| c6i.2xlarge | 8 | 8 | 32 |
| c6i.4xlarge | 8 | 16 | 64 |
| c6i.8xlarge | 8 | 32 | 64 |
| c6i.12xlarge | 8 | 32 | 64 |
| c6i.16xlarge | 8 | 32 | 120 |
| c6i.24xlarge | 8 | 32 | 120 |
| c6i.32xlarge | 8 | 32 | 120 |
| c6id.large | 2 | 2 | 6 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c6id.xlarge | 4 | 4 | 16 |
| c6id.2xlarge | 8 | 8 | 32 |
| c6id.4xlarge | 8 | 16 | 64 |
| c6id.8xlarge | 8 | 32 | 64 |
| c6id.12xlarge | 8 | 32 | 64 |
| c6id.16xlarge | 8 | 32 | 120 |
| c6id.24xlarge | 8 | 32 | 120 |
| c6id.32xlarge | 8 | 32 | 120 |
| c6in.large | 2 | 2 | 6 |
| c6in.xlarge | 4 | 4 | 16 |
| c6in.2xlarge | 8 | 8 | 32 |
| c6in.4xlarge | 8 | 16 | 64 |
| c6in.8xlarge | 16 | 32 | 128 |
| c6in.12xlarge | 16 | 32 | 128 |
| c6in.16xlarge | 16 | 32 | 240 |
| c6in.24xlarge | 32 | 32 | 480 |
| c6in.32xlarge | 32 | 32 | 512* |
| c8a.medium | 1 | 1 | 3 |
| c8a.large | 2 | 2 | 6 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c8a.xlarge | 4 | 4 | 16 |
| c8a.2xlarge | 8 | 8 | 32 |
| c8a.4xlarge | 8 | 16 | 64 |
| c8a.8xlarge | 8 | 32 | 128 |
| c8a.12xlarge | 16 | 64 | 192 |
| c8a.16xlarge | 16 | 64 | 256 |
| c8a.24xlarge | 16 | 128 | 384 |
| c8a.48xlarge | 32 | 128 | 768 |
| c8a.metal-24xl | 16 | 128 | 384 |
| c8a.metal-48xl | 32 | 128 | 768 |
| c8gb.medium | 1 | 1 | 2 |
| c8gb.large | 2 | 2 | 6 |
| c8gb.xlarge | 4 | 4 | 16 |
| c8gb.2xlarge | 8 | 8 | 32 |
| c8gb.4xlarge | 8 | 16 | 64 |
| c8gb.8xlarge | 8 | 32 | 128 |
| c8gb.12xlarge | 16 | 64 | 192 |
| c8gb.16xlarge | 16 | 64 | 256 |
| c8gb.24xlarge | 16 | 128 | 384 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c8gb.48xlarge | 32 | 128 | 768* |
| c8gb.metal-24xl | 32 | 128 | 768 |
| c8gb.metal-48xl | 32 | 128 | 768* |
| c8gn.medium | 1 | 1 | 2 |
| c8gn.large | 2 | 2 | 6 |
| c8gn.xlarge | 4 | 4 | 16 |
| c8gn.2xlarge | 8 | 8 | 32 |
| c8gn.4xlarge | 8 | 16 | 64 |
| c8gn.8xlarge | 8 | 32 | 128 |
| c8gn.12xlarge | 16 | 64 | 192 |
| c8gn.16xlarge | 16 | 64 | 256 |
| c8gn.24xlarge | 16 | 128 | 384 |
| c8gn.48xlarge | 32 | 128 | 768* |
| c8gn.metal-24xl | 32 | 128 | 768 |
| c8gn.metal-48xl | 32 | 128 | 768* |
| c8i.large | 2 | 2 | 6 |
| c8i.xlarge | 4 | 4 | 16 |
| c8i.2xlarge | 8 | 8 | 32 |
| c8i.4xlarge | 8 | 16 | 64 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c8i.8xlarge | 8 | 32 | 128 |
| c8i.12xlarge | 16 | 64 | 192 |
| c8i.16xlarge | 16 | 64 | 256 |
| c8i.24xlarge | 16 | 128 | 384 |
| c8i.32xlarge | 16 | 128 | 512 |
| c8i.48xlarge | 32 | 128 | 768 |
| c8i.96xlarge | 32 | 128 | 1536 |
| c8i.metal-48xl | 32 | 128 | 768 |
| c8i.metal-96xl | 32 | 128 | 1536 |
| c8id.large | 2 | 2 | 6 |
| c8id.xlarge | 4 | 4 | 16 |
| c8id.2xlarge | 8 | 8 | 32 |
| c8id.4xlarge | 8 | 16 | 64 |
| c8id.8xlarge | 8 | 32 | 128 |
| c8id.12xlarge | 16 | 64 | 192 |
| c8id.16xlarge | 16 | 64 | 256 |
| c8id.24xlarge | 16 | 128 | 384 |
| c8id.32xlarge | 16 | 128 | 512 |
| c8id.48xlarge | 32 | 128 | 768 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c8id.96xlarge | 32 | 128 | 1536 |
| c8id.metal-48xl | 32 | 128 | 768 |
| c8id.metal-96xl | 32 | 128 | 1536 |
| c8i-flex.large | 1 | 1 | 3 |
| c8i-flex.xlarge | 2 | 2 | 8 |
| c8i-flex.2xlarge | 4 | 4 | 16 |
| c8i-flex.4xlarge | 4 | 8 | 32 |
| c8i-flex.8xlarge | 4 | 16 | 64 |
| c8i-flex.12xlarge | 8 | 32 | 96 |
| c8i-flex.16xlarge | 8 | 32 | 128 |
| c8in.large | 2 | 2 | 8 |
| c8in.xlarge | 4 | 4 | 16 |
| c8in.2xlarge | 8 | 8 | 32 |
| c8in.4xlarge | 8 | 16 | 64 |
| c8in.8xlarge | 16 | 32 | 128 |
| c8in.12xlarge | 16 | 64 | 192 |
| c8in.16xlarge | 16 | 64 | 256 |
| c8in.24xlarge | 16 | 128 | 256 |
| c8in.32xlarge | 32 | 128 | 512 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c8in.48xlarge | 32 | 128 | 768 |
| c8in.96xlarge | 32 | 128 | 1536* |
| c8in.metal-48xl | 32 | 128 | 768 |
| c8in.metal-96xl | 32 | 128 | 1536* |
| c8ine.large | 2 | 2 | 8 |
| c8ine.xlarge | 4 | 4 | 16 |
| c8ine.2xlarge | 8 | 8 | 32 |
| c8ine.4xlarge | 16 | 16 | 128 |
| c8ine.8xlarge | 32 | 32 | 256 |
| c8ine.12xlarge | 32 | 64 | 384 |
| c8ib.large | 2 | 2 | 8 |
| c8ib.xlarge | 4 | 4 | 16 |
| c8ib.2xlarge | 8 | 8 | 32 |
| c8ib.4xlarge | 8 | 16 | 64 |
| c8ib.8xlarge | 16 | 32 | 128 |
| c8ib.12xlarge | 16 | 64 | 192 |
| c8ib.16xlarge | 16 | 64 | 256 |
| c8ib.24xlarge | 16 | 128 | 256 |
| c8ib.32xlarge | 32 | 128 | 512 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c8ib.48xlarge | 32 | 128 | 768 |
| c8ib.96xlarge | 32 | 128 | 1536* |
| c8ib.metal-48xl | 32 | 128 | 768 |
| c8ib.metal-96xl | 32 | 128 | 1536* |
| c9g.medium | 1 | 1 | 2 |
| c9g.large | 2 | 2 | 6 |
| c9g.xlarge | 4 | 4 | 16 |
| c9g.2xlarge | 8 | 8 | 32 |
| c9g.4xlarge | 8 | 16 | 64 |
| c9g.8xlarge | 8 | 32 | 128 |
| c9g.12xlarge | 16 | 64 | 192 |
| c9g.16xlarge | 16 | 64 | 256 |
| c9g.24xlarge | 16 | 128 | 384 |
| c9g.48xlarge | 32 | 128 | 768 |
| c9g.metal-48xl | 32 | 128 | 768 |
| c9gd.medium | 1 | 1 | 2 |
| c9gd.large | 2 | 2 | 6 |
| c9gd.xlarge | 4 | 4 | 16 |
| c9gd.2xlarge | 8 | 8 | 32 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| c9gd.4xlarge | 8 | 16 | 64 |
| c9gd.8xlarge | 8 | 32 | 128 |
| c9gd.12xlarge | 16 | 64 | 192 |
| c9gd.16xlarge | 16 | 64 | 256 |
| c9gd.24xlarge | 16 | 128 | 384 |
| c9gd.48xlarge | 32 | 128 | 768 |
| c9gd.metal-48xl | 32 | 128 | 768 |

Memory optimized

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r6i.large | 2 | 2 | 6 |
| r6i.xlarge | 4 | 4 | 16 |
| r6i.2xlarge | 8 | 8 | 32 |
| r6i.4xlarge | 8 | 16 | 64 |
| r6i.8xlarge | 8 | 32 | 64 |
| r6i.12xlarge | 8 | 32 | 64 |
| r6i.16xlarge | 8 | 32 | 120 |
| r6i.24xlarge | 8 | 32 | 120 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r6i.32xlarge | 8 | 32 | 120 |
| r6id.large | 2 | 2 | 6 |
| r6id.xlarge | 4 | 4 | 16 |
| r6id.2xlarge | 8 | 8 | 32 |
| r6id.4xlarge | 8 | 16 | 64 |
| r6id.8xlarge | 8 | 32 | 64 |
| r6id.12xlarge | 8 | 32 | 64 |
| r6id.16xlarge | 8 | 32 | 120 |
| r6id.24xlarge | 8 | 32 | 120 |
| r6id.32xlarge | 8 | 32 | 120 |
| r6idn.large | 2 | 2 | 6 |
| r6idn.xlarge | 4 | 4 | 16 |
| r6idn.2xlarge | 8 | 8 | 32 |
| r6idn.4xlarge | 8 | 16 | 64 |
| r6idn.8xlarge | 16 | 32 | 128 |
| r6idn.12xlarge | 16 | 32 | 128 |
| r6idn.16xlarge | 16 | 32 | 240 |
| r6idn.24xlarge | 32 | 32 | 480 |
| r6idn.32xlarge | 32 | 32 | 512* |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r6in.large | 2 | 2 | 6 |
| r6in.xlarge | 4 | 4 | 16 |
| r6in.2xlarge | 8 | 8 | 32 |
| r6in.4xlarge | 8 | 16 | 64 |
| r6in.8xlarge | 16 | 32 | 128 |
| r6in.12xlarge | 16 | 32 | 128 |
| r6in.16xlarge | 16 | 32 | 240 |
| r6in.24xlarge | 32 | 32 | 480 |
| r6in.32xlarge | 32 | 32 | 512* |
| r8a.medium | 1 | 1 | 3 |
| r8a.large | 2 | 2 | 6 |
| r8a.xlarge | 4 | 4 | 16 |
| r8a.2xlarge | 8 | 8 | 32 |
| r8a.4xlarge | 8 | 16 | 64 |
| r8a.8xlarge | 8 | 32 | 128 |
| r8a.12xlarge | 16 | 64 | 192 |
| r8a.16xlarge | 16 | 64 | 256 |
| r8a.24xlarge | 16 | 128 | 384 |
| r8a.48xlarge | 32 | 128 | 768 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r8a.metal-24xl | 16 | 128 | 384 |
| r8a.metal-48xl | 32 | 128 | 768 |
| r8gb.medium | 1 | 1 | 2 |
| r8gb.large | 2 | 2 | 6 |
| r8gb.xlarge | 4 | 4 | 16 |
| r8gb.2xlarge | 8 | 8 | 32 |
| r8gb.4xlarge | 8 | 16 | 64 |
| r8gb.8xlarge | 8 | 32 | 128 |
| r8gb.12xlarge | 16 | 64 | 192 |
| r8gb.16xlarge | 16 | 64 | 256 |
| r8gb.24xlarge | 16 | 128 | 384 |
| r8gb.48xlarge | 32 | 128 | 768* |
| r8gb.metal-24xl | 32 | 128 | 768 |
| r8gb.metal-48xl | 32 | 128 | 768* |
| r8gn.medium | 1 | 1 | 2 |
| r8gn.large | 2 | 2 | 6 |
| r8gn.xlarge | 4 | 4 | 16 |
| r8gn.2xlarge | 8 | 8 | 32 |
| r8gn.4xlarge | 8 | 16 | 64 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r8gn.8xlarge | 8 | 32 | 128 |
| r8gn.12xlarge | 16 | 64 | 192 |
| r8gn.16xlarge | 16 | 64 | 256 |
| r8gn.24xlarge | 16 | 128 | 384 |
| r8gn.48xlarge | 32 | 128 | 768* |
| r8gn.metal-24xl | 32 | 128 | 768 |
| r8gn.metal-48xl | 32 | 128 | 768* |
| r8i.large | 2 | 2 | 6 |
| r8i.xlarge | 4 | 4 | 16 |
| r8i.2xlarge | 8 | 8 | 32 |
| r8i.4xlarge | 8 | 16 | 64 |
| r8i.8xlarge | 8 | 32 | 128 |
| r8i.12xlarge | 16 | 64 | 192 |
| r8i.16xlarge | 16 | 64 | 256 |
| r8i.24xlarge | 16 | 128 | 384 |
| r8i.32xlarge | 16 | 128 | 512 |
| r8i.48xlarge | 32 | 128 | 768 |
| r8i.96xlarge | 32 | 128 | 1536 |
| r8i.metal-48xl | 32 | 128 | 768 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r8i.metal-96xl | 32 | 128 | 1536 |
| r8id.large | 2 | 2 | 6 |
| r8id.xlarge | 4 | 4 | 16 |
| r8id.2xlarge | 8 | 8 | 32 |
| r8id.4xlarge | 8 | 16 | 64 |
| r8id.8xlarge | 8 | 32 | 128 |
| r8id.12xlarge | 16 | 64 | 192 |
| r8id.16xlarge | 16 | 64 | 256 |
| r8id.24xlarge | 16 | 128 | 384 |
| r8id.32xlarge | 16 | 128 | 512 |
| r8id.48xlarge | 32 | 128 | 768 |
| r8id.96xlarge | 32 | 128 | 1536 |
| r8id.metal-48xl | 32 | 128 | 768 |
| r8id.metal-96xl | 32 | 128 | 1536 |
| r8i-flex.large | 1 | 1 | 3 |
| r8i-flex.xlarge | 2 | 2 | 8 |
| r8i-flex.2xlarge | 4 | 4 | 16 |
| r8i-flex.4xlarge | 4 | 8 | 32 |
| r8i-flex.8xlarge | 4 | 16 | 64 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r8i-flex.12xlarge | 8 | 32 | 96 |
| r8i-flex.16xlarge | 8 | 32 | 128 |
| r8in.large | 2 | 2 | 8 |
| r8in.xlarge | 4 | 4 | 16 |
| r8in.2xlarge | 8 | 8 | 32 |
| r8in.4xlarge | 8 | 16 | 64 |
| r8in.8xlarge | 16 | 32 | 128 |
| r8in.12xlarge | 16 | 64 | 192 |
| r8in.16xlarge | 16 | 64 | 256 |
| r8in.24xlarge | 16 | 128 | 256 |
| r8in.32xlarge | 32 | 128 | 512 |
| r8in.48xlarge | 32 | 128 | 768 |
| r8in.96xlarge | 32 | 128 | 1536* |
| r8in.metal-48xl | 32 | 128 | 768 |
| r8in.metal-96xl | 32 | 128 | 1536* |
| r8idn.large | 2 | 2 | 8 |
| r8idn.xlarge | 4 | 4 | 16 |
| r8idn.2xlarge | 8 | 8 | 32 |
| r8idn.4xlarge | 8 | 16 | 64 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r8idn.8xlarge | 16 | 32 | 128 |
| r8idn.12xlarge | 16 | 64 | 192 |
| r8idn.16xlarge | 16 | 64 | 256 |
| r8idn.24xlarge | 16 | 128 | 256 |
| r8idn.32xlarge | 32 | 128 | 512 |
| r8idn.48xlarge | 32 | 128 | 768 |
| r8idn.96xlarge | 32 | 128 | 1536* |
| r8idn.metal-48xl | 32 | 128 | 768 |
| r8idn.metal-96xl | 32 | 128 | 1536* |
| r8ib.large | 2 | 2 | 8 |
| r8ib.xlarge | 4 | 4 | 16 |
| r8ib.2xlarge | 8 | 8 | 32 |
| r8ib.4xlarge | 8 | 16 | 64 |
| r8ib.8xlarge | 16 | 32 | 128 |
| r8ib.12xlarge | 16 | 64 | 192 |
| r8ib.16xlarge | 16 | 64 | 256 |
| r8ib.24xlarge | 16 | 128 | 256 |
| r8ib.32xlarge | 32 | 128 | 512 |
| r8ib.48xlarge | 32 | 128 | 768 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r8ib.96xlarge | 32 | 128 | 1536* |
| r8ib.metal-48xl | 32 | 128 | 768 |
| r8ib.metal-96xl | 32 | 128 | 1536* |
| r8idb.large | 2 | 2 | 8 |
| r8idb.xlarge | 4 | 4 | 16 |
| r8idb.2xlarge | 8 | 8 | 32 |
| r8idb.4xlarge | 8 | 16 | 64 |
| r8idb.8xlarge | 16 | 32 | 128 |
| r8idb.12xlarge | 16 | 64 | 192 |
| r8idb.16xlarge | 16 | 64 | 256 |
| r8idb.24xlarge | 16 | 128 | 256 |
| r8idb.32xlarge | 32 | 128 | 512 |
| r8idb.48xlarge | 32 | 128 | 768 |
| r8idb.96xlarge | 32 | 128 | 1536* |
| r8idb.metal-48xl | 32 | 128 | 768 |
| r8idb.metal-96xl | 32 | 128 | 1536* |
| r9g.medium | 1 | 1 | 2 |
| r9g.large | 2 | 2 | 6 |
| r9g.xlarge | 4 | 4 | 16 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| r9g.2xlarge | 8 | 8 | 32 |
| r9g.4xlarge | 8 | 16 | 64 |
| r9g.8xlarge | 8 | 32 | 128 |
| r9g.12xlarge | 16 | 64 | 192 |
| r9g.16xlarge | 16 | 64 | 256 |
| r9g.24xlarge | 16 | 128 | 384 |
| r9g.48xlarge | 32 | 128 | 768 |
| r9g.metal-48xl | 32 | 128 | 768 |
| r9gd.medium | 1 | 1 | 2 |
| r9gd.large | 2 | 2 | 6 |
| r9gd.xlarge | 4 | 4 | 16 |
| r9gd.2xlarge | 8 | 8 | 32 |
| r9gd.4xlarge | 8 | 16 | 64 |
| r9gd.8xlarge | 8 | 32 | 128 |
| r9gd.12xlarge | 16 | 64 | 192 |
| r9gd.16xlarge | 16 | 64 | 256 |
| r9gd.24xlarge | 16 | 128 | 384 |
| r9gd.48xlarge | 32 | 128 | 768 |
| r9gd.metal-48xl | 32 | 128 | 768 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| x8aedz.large | 2 | 2 | 8 |
| x8aedz.xlarge | 4 | 4 | 16 |
| x8aedz.3xlarge | 4 | 16 | 48 |
| x8aedz.6xlarge | 8 | 32 | 96 |
| x8aedz.12xlarge | 8 | 64 | 192 |
| x8aedz.24xlarge | 16 | 128 | 384 |
| x8aedz.metal-12xl | 8 | 64 | 192 |
| x8aedz.metal-24xl | 16 | 128 | 384 |
| x8i.large | 2 | 2 | 6 |
| x8i.xlarge | 4 | 4 | 16 |
| x8i.2xlarge | 8 | 8 | 32 |
| x8i.4xlarge | 8 | 16 | 64 |
| x8i.8xlarge | 8 | 32 | 128 |
| x8i.12xlarge | 16 | 64 | 192 |
| x8i.16xlarge | 16 | 64 | 256 |
| x8i.24xlarge | 16 | 128 | 384 |
| x8i.32xlarge | 16 | 128 | 512 |
| x8i.48xlarge | 32 | 128 | 768 |
| x8i.64xlarge | 32 | 128 | 1024 |

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| x8i.96xlarge | 32 | 128 | 1536 |
| x8i.metal-48xl | 32 | 128 | 768 |
| x8i.metal-96xl | 32 | 128 | 1536 |

Storage optimized

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| i8ge.large | 2 | 2 | 6 |
| i8ge.xlarge | 4 | 4 | 16 |
| i8ge.2xlarge | 8 | 8 | 32 |
| i8ge.3xlarge | 8 | 16 | 48 |
| i8ge.6xlarge | 8 | 32 | 96 |
| i8ge.12xlarge | 16 | 64 | 192 |
| i8ge.18xlarge | 16 | 128 | 288 |
| i8ge.24xlarge | 16 | 128 | 384 |
| i8ge.48xlarge | 32 | 128 | 1536* |

Accelerated computing

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| g7.2xlarge | 8 | 8 | 32 |
| g7.4xlarge | 8 | 16 | 64 |
| g7.8xlarge | 8 | 32 | 80 |
| g7.12xlarge | 16 | 64 | 192 |
| g7.24xlarge | 16 | 128 | 384 |
| g7.48xlarge | 32 | 128 | 768* |

High performance computing

| Tipe instans | Antrian ENA default per antarmuka | Antrian ENA maksimum per antarmuka | Antrian ENA maksimum per instans |
|---|---|---|---|
| hpc8a.96xlarge | 32 | 128 | 512* |

Note

* Jenis instance ini menampilkan beberapa kartu jaringan. Jenis instance lainnya menampilkan kartu jaringan tunggal. Untuk informasi selengkapnya, lihat Kartu jaringan.

Konfigurasikan alokasi antrian ENA

Pada jenis instans yang didukung, Anda dapat mengonfigurasi jumlah antrian ENA untuk setiap ENI. Gunakan Konsol Manajemen AWS, AWS CLI, atau PowerShell. Di Konsol Manajemen AWS, pengaturan ini muncul di bawah setiap antarmuka  Jaringan.

> **Note**
>
> Sebelum Anda mengonfigurasi antrian ENA, perhatikan persyaratan berikut:
>
> - Instans Anda harus dihentikan sebelum Anda mengonfigurasi jumlah antrian ENA.
> - Nilai untuk antrian ENA harus pangkat 2, seperti 1, 2, 4, 8, 16, atau 32.
> - Jumlah antrian pada ENI tunggal mana pun tidak boleh melebihi jumlah vCPU pada instans Anda.

Sebelum Anda mengubah jumlah antrian, periksa hitungan Anda saat ini dengan perintah berikut.

AWS CLI

```
aws ec2 describe-instances --instance-id i-1234567890abcdef0
```

PowerShell

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances.NetworkInterfaces |
  Select-Object NetworkInterfaceId,
    @{N='DeviceIndex';E={$_.Attachment.DeviceIndex}},
    @{N='AttachmentId';E={$_.Attachment.AttachmentId}},
    @{N='EnaQueueCount';E={$_.Attachment.EnaQueueCount}}
```

Melampirkan antarmuka jaringan dengan antrian ENA

Dalam contoh berikut, 16 antrian ENA dikonfigurasi pada ENI.

AWS CLI

Untuk melampirkan antarmuka jaringan dengan antrian ENA

Gunakan perintah attach-network-interface.

```
aws ec2 attach-network-interface \
  --network-interface-id eni-abcdef01234567890 \
  --instance-id i-1234567890abcdef0 \
  --device-index 1 \
  --ena-queue-count 16
```

PowerShell

Untuk melampirkan antarmuka jaringan dengan antrian ENA

Gunakan Add-EC2NetworkInterface cmdlet.

```
Add-EC2NetworkInterface `
  -NetworkInterfaceId eni-abcdef01234567890 `
  -InstanceId i-1234567890abcdef0 `
  -DeviceIndex 1 `
  -EnaQueueCount 16
```

Luncurkan instance dengan antrian ENA

Dalam contoh berikut, 16 antrian ENA masing-masing dikonfigurasi pada 3 ENI.

AWS CLI

Untuk meluncurkan instance dengan antrian ENA

Gunakan perintah run-instans.

```
aws ec2 run-instances \
  --image-id ami-1234567890abcdef0 \
  --instance-type c8i.4xlarge \
  --network-interfaces \
    "[{\"DeviceIndex\":0,\"SubnetId\":\"subnet-abcdef01234567890\",\"EnaQueueCount
\":16},
      {\"DeviceIndex\":1,\"SubnetId\":\"subnet-abcdef01234567890\",\"EnaQueueCount
\":16},
      {\"DeviceIndex\":2,\"SubnetId\":\"subnet-abcdef01234567890\",\"EnaQueueCount
\":16}]"
```

PowerShell

Untuk meluncurkan instance dengan antrian ENA

Gunakan New-EC2Instance cmdlet.

```
New-EC2Instance `
  -ImageId ami-1234567890abcdef0 `
  -InstanceType c8i.4xlarge `
  -NetworkInterface @(
```

```
    @{DeviceIndex=0; SubnetId="subnet-abcdef01234567890"; EnaQueueCount=16},
    @{DeviceIndex=1; SubnetId="subnet-abcdef01234567890"; EnaQueueCount=16},
    @{DeviceIndex=2; SubnetId="subnet-abcdef01234567890"; EnaQueueCount=16}
  )
```

Memodifikasi antrian ENA pada antarmuka jaringan yang ada

Dalam contoh berikut, 16 antrian ENA dikonfigurasi pada ENI.

AWS CLI

Untuk memodifikasi antrian ENA pada antarmuka jaringan

Gunakan perintah modify- network-interface-atribut.

```
aws ec2 modify-network-interface-attribute \
  --network-interface-id eni-1234567890abcdef0 \
  --attachment AttachmentId=eni-attach-1234567890abcdef0,EnaQueueCount=16
```

PowerShell

Untuk memodifikasi antrian ENA pada antarmuka jaringan

Gunakan Edit-EC2NetworkInterfaceAttribute cmdlet.

```
Edit-EC2NetworkInterfaceAttribute `
  -NetworkInterfaceId eni-1234567890abcdef0 `
  -Attachment_AttachmentId eni-attach-1234567890abcdef0 `
  -Attachment_EnaQueueCount 16
```

Dalam contoh berikut, jumlah ENA diatur ulang ke nilai default.

AWS CLI

```
aws ec2 modify-network-interface-attribute \
  --network-interface-id eni-1234567890abcdef0 \
  --attachment AttachmentId=eni-attach-1234567890abcdef0,DefaultEnaQueueCount=true
```

PowerShell

```
Edit-EC2NetworkInterfaceAttribute `
```

```
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -Attachment_AttachmentId eni-attach-1234567890abcdef0 `
    -Attachment_DefaultEnaQueueCount $true
```

## Memecahkan masalah driver kernel ENA di Linux

Adaptor Jaringan Elastis (ENA) dirancang untuk meningkatkan kesehatan sistem operasi dan mengurangi kemungkinan gangguan jangka panjang karena perilaku dan atau kegagalan perangkat keras yang tidak terduga. Arsitektur ENA menjaga kegagalan perangkat atau driver setransparan mungkin ke sistem. Topik ini memberikan informasi pemecahan masalah untuk ENA.

Jika Anda tidak dapat terhubung ke instans, mulailah dengan bagian Memecahkan masalah konektivitas.

Jika Anda mengalami penurunan kinerja setelah bermigrasi ke jenis instans generasi keenam, lihat artikel Apa yang harus saya lakukan sebelum memigrasikan instans EC2 saya ke instans generasi keenam untuk memastikan bahwa saya mendapatkan kinerja jaringan maksimum?

Jika Anda dapat menyambungkan ke instans, Anda dapat mengumpulkan informasi diagnostik dengan menggunakan deteksi kegagalan dan mekanisme pemulihan yang dibahas di bagian selanjutnya dari topik ini.

Daftar Isi


- Memecahkan masalah konektivitas
- Keep-alive mekanisme
- Mendaftarkan waktu habis baca
- Statistik
- Log error driver di syslog
- Sub-optimal pemberitahuan konfigurasi


### Memecahkan masalah konektivitas

Jika Anda kehilangan konektivitas saat mengaktifkan jaringan yang ditingkatkan, modul `ena` mungkin tidak kompatibel dengan kernel yang sedang berjalan dari instans Anda. Ini dapat terjadi jika Anda menginstal modul untuk versi kernel tertentu (tanpa dkms, atau dengan file dkms.conf yang tidak dikonfigurasi dengan benar) lalu kernel instans Anda diperbarui. Jika kernel instans yang dimuat

saat boot tidak memiliki modul `ena` yang diinstal dengan benar, instans Anda tidak akan mengenali adaptor jaringan dan instans Anda tidak dapat dijangkau.

Jika Anda mengaktifkan jaringan yang ditingkatkan untuk instans PV atau AMI, ini juga dapat membuat instans Anda tidak dapat dijangkau.

Jika instans Anda tidak dapat dijangkau setelah mengaktifkan jaringan yang ditingkatkan dengan ENA, Anda dapat menonaktifkan atribut `enaSupport` untuk instans Anda dan atribut itu akan kembali ke adaptor jaringan stok.

Untuk menonaktifkan jaringan yang ditingkatkan dengan ENA (EBS-backed instance)

1. Dari komputer lokal Anda, hentikan instance dengan menggunakan konsol Amazon EC2, perintah stop-instance (AWS CLI), atau Stop-EC2Instance cmdlet ().Alat AWS untuk PowerShell
2. Dari komputer lokal Anda, nonaktifkan atribut jaringan yang disempurnakan dengan menggunakan perintah modify-instance-atribut dengan `--no-ena-support` opsi atau cmdlet dengan parameter. Edit-EC2InstanceAttribute `-EnaSupport $false`
3. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2, perintah start-instance, atau cmdlet. Start-EC2Instance
4. (Opsional) Connect ke instans Anda dan coba instal ulang modul `ena` dengan versi kernel Anda saat ini, dengan mengikuti langkah-langkah di Aktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda.

Untuk menonaktifkan jaringan yang ditingkatkan dengan ENA (instans yang didukung penyimpanan instans)

1. Buat AMI baru seperti yang dijelaskan diBuat S3-backed AMI Amazon.
2. Saat Anda mendaftarkan AMI, pastikan untuk menyertakan `--no-ena-support` opsi dalam perintah stop-instance (AWS CLI) atau `-EnaSupport $false` parameter dalam cmdlet. Register-EC2Image

Keep-alive mekanisme

Perangkat ENA memposting peristiwa keep-alive dengan kecepatan tetap (biasanya sekali setiap detik). Driver ENA mengimplementasikan mekanisme watchdog, yang memeriksa keberadaan pesan

keep-alive ini. Jika ada pesan atau beberapa pesan, watchdog disiapkan kembali, jika tidak driver menyimpulkan bahwa perangkat mengalami kegagalan dan kemudian melakukan hal berikut:

- Membuang statistik saat ini ke syslog
- Mereset perangkat ENA
- Mereset status driver ENA

Prosedur reset di atas dapat mengakibatkan kehilangan beberapa lalu lintas dalam waktu singkat (koneksi TCP harus dapat dipulihkan), tetapi seharusnya tidak mempengaruhi pengguna.

Perangkat ENA juga secara tidak langsung dapat meminta prosedur reset perangkat, misalnya dengan tidak mengirimkan pemberitahuan keep-alive, jika perangkat ENA mencapai status tidak diketahui setelah memuat konfigurasi yang tidak dapat dipulihkan.

Berikut ini adalah contoh prosedur reset:

```
[18509.800135] ena 0000:00:07.0 eth1: Keep alive watchdog timeout. // The watchdog
 process initiates a reset
[18509.815244] ena 0000:00:07.0 eth1: Trigger reset is on
[18509.825589] ena 0000:00:07.0 eth1: tx_timeout: 0 // The driver logs the current
 statistics
[18509.834253] ena 0000:00:07.0 eth1: io_suspend: 0
[18509.842674] ena 0000:00:07.0 eth1: io_resume: 0
[18509.850275] ena 0000:00:07.0 eth1: wd_expired: 1
[18509.857855] ena 0000:00:07.0 eth1: interface_up: 1
[18509.865415] ena 0000:00:07.0 eth1: interface_down: 0
[18509.873468] ena 0000:00:07.0 eth1: admin_q_pause: 0
[18509.881075] ena 0000:00:07.0 eth1: queue_0_tx_cnt: 0
[18509.888629] ena 0000:00:07.0 eth1: queue_0_tx_bytes: 0
[18509.895286] ena 0000:00:07.0 eth1: queue_0_tx_queue_stop: 0
.......
........
[18511.280972] ena 0000:00:07.0 eth1: free uncompleted tx skb qid 3 idx 0x7 // At the
 end of the down process, the driver discards incomplete packets.
[18511.420112] [ENA_COM: ena_com_validate_version] ena device version: 0.10 //The
 driver begins its up process
[18511.420119] [ENA_COM: ena_com_validate_version] ena controller version: 0.0.1
 implementation version 1
[18511.420127] [ENA_COM: ena_com_admin_init] ena_defs : Version:[b9692e8] Build date
 [Wed Apr  6 09:54:21 IDT 2016]
[18512.252108] ena 0000:00:07.0: Device watchdog is Enabled
```

```
[18512.674877] ena 0000:00:07.0: irq 46 for MSI/MSI-X
[18512.674933] ena 0000:00:07.0: irq 47 for MSI/MSI-X
[18512.674990] ena 0000:00:07.0: irq 48 for MSI/MSI-X
[18512.675037] ena 0000:00:07.0: irq 49 for MSI/MSI-X
[18512.675085] ena 0000:00:07.0: irq 50 for MSI/MSI-X
[18512.675141] ena 0000:00:07.0: irq 51 for MSI/MSI-X
[18512.675188] ena 0000:00:07.0: irq 52 for MSI/MSI-X
[18512.675233] ena 0000:00:07.0: irq 53 for MSI/MSI-X
[18512.675279] ena 0000:00:07.0: irq 54 for MSI/MSI-X
[18512.772641] [ENA_COM: ena_com_set_hash_function] Feature 10 isn't supported
[18512.772647] [ENA_COM: ena_com_set_hash_ctrl] Feature 18 isn't supported
[18512.775945] ena 0000:00:07.0: Device reset completed successfully // The reset
 process is complete
```

Mendaftarkan waktu habis baca

Arsitektur ENA menyarankan penggunaan terbatas operasi pemetaan memori I/O (MMIO) baca. Daftar MMIO diakses oleh pengandar perangkat ENA hanya selama prosedur inisialisasi.

Jika log driver (tersedia dalam output dmesg) menunjukkan kegagalan operasi baca, ini mungkin disebabkan oleh driver yang tidak kompatibel atau tidak dikompilasi dengan benar, hardware yang sibuk, atau kegagalan hardware.

Entri log terputus-putus yang menunjukkan kegagalan pada operasi baca tidak boleh dianggap sebagai masalah; dalam kasus ini driver akan mencobanya kembali. Namun, urutan entri log yang berisi kegagalan pembacaan menunjukkan masalah driver atau hardware.

Di bawah ini adalah contoh entri log driver yang menunjukkan kegagalan operasi baca karena waktu habis:

```
[ 47.113698] [ENA_COM: ena_com_reg_bar_read32] reading reg failed for timeout.
 expected: req id[1] offset[88] actual: req id[57006] offset[0]
[ 47.333715] [ENA_COM: ena_com_reg_bar_read32] reading reg failed for timeout.
 expected: req id[2] offset[8] actual: req id[57007] offset[0]
[ 47.346221] [ENA_COM: ena_com_dev_reset] Reg read32 timeout occurred
```

Statistik

Jika Anda mengalami masalah latensi atau performa jaringan yang tidak memadai, Anda harus mengambil statistik perangkat dan memeriksanya. Statistik ini dapat diperoleh dengan menggunakan ethtool, seperti berikut ini.

```
[ec2-user ~]$ ethtool -S ethN
NIC statistics:
 tx_timeout: 0
 suspend: 0
 resume: 0
 wd_expired: 0
 interface_up: 1
 interface_down: 0
 admin_q_pause: 0
 bw_in_allowance_exceeded: 0
 bw_out_allowance_exceeded: 0
 pps_allowance_exceeded: 0
 conntrack_allowance_available: 450878
 conntrack_allowance_exceeded: 0
 linklocal_allowance_exceeded: 0
 queue_0_tx_cnt: 4329
 queue_0_tx_bytes: 1075749
 queue_0_tx_queue_stop: 0
 ...
```

Parameter output perintah berikut dijelaskan di bawah ini:

`tx_timeout`: *N*

Berapa kali watchdog Netdev diaktifkan.

`suspend`: *N*

Jumlah berapa kali driver melakukan operasi penangguhan.

`resume`: *N*

Jumlah berapa kali driver melakukan operasi kelanjutan.

`wd_expired`: *N*

Berapa kali driver tersebut tidak menerima peristiwa keep-alive dalam tiga detik sebelumnya.

`interface_up`: *N*

Berapa kali antarmuka ENA dinaikkan.

`interface_down`: *N*

Berapa kali antarmuka ENA diturunkan.

`admin_q_pause`: *N*

Jumlah berapa kali antrean admin tidak ditemukan dalam status berjalan.

`bw_in_allowance_exceeded`: *N*

Jumlah paket antri atau dijatuhkan karena kumpulan bandwidth yang masuk melebihi maksimum untuk instans.

`bw_out_allowance_exceeded`: *N*

Jumlah paket antri atau dijatuhkan karena kumpulan bandwidth agregat yang keluar melebihi maksimum untuk instans.

`pps_allowance_exceeded`: *N*

Jumlah paket yang mengantri atau dijatuhkan karena PPS dua arah melebihi maksimum untuk instance. *

`conntrack_allowance_available`: *N*

Jumlah koneksi yang dilacak yang dapat dibuat oleh instans sebelum menekan tunjangan Connections Tracked dari tipe instans tersebut. Hanya tersedia untuk Nitro-based contoh. Tidak didukung dengan FreeBSD instans atau lingkungan DPDK.

`conntrack_allowance_exceeded`: *N*

Jumlah paket turun karena pelacakan koneksi melebihi maksimum untuk instans dan koneksi baru tidak dapat dibuat. Hal ini dapat mengakibatkan hilangnya paket untuk lalu lintas ke atau dari instans.

`linklocal_allowance_exceeded`: *N*

Jumlah paket turun karena PPS lalu lintas ke layanan proksi lokal melebihi batas maksimum untuk antarmuka jaringan. Ini berdampak pada lalu lintas ke layanan Amazon DNS, Layanan Metadata Instans, dan Layanan Sinkronisasi Waktu Amazon, tetapi tidak memengaruhi lalu lintas ke resolver DNS khusus.

`queue_`*N*`_tx_cnt`: *N*

Jumlah paket yang dikirimkan untuk antrean ini.

`queue_`*N*`_tx_bytes`: *N*

Jumlah bita yang dikirimkan untuk antrean ini.

`queue_N_tx_queue_stop`: *N*

Berapa kali antrian *N* itu penuh dan berhenti.

`queue_N_tx_queue_wakeup`: *N*

Jumlah kali antrian *N* dilanjutkan setelah dihentikan.

`queue_N_tx_dma_mapping_err`: *N*

Jumlah kesalahan akses memori langsung. Jika nilai ini bukan 0, ini menunjukkan sumber daya sistem yang rendah.

`queue_N_tx_linearize`: *N*

Jumlah berapa kali linierisasi SKB dicoba untuk antrean.

`queue_N_tx_linearize_failed`: *N*

Jumlah berapa kali linierisasi SKB gagal untuk antrean ini.

`queue_N_tx_napi_comp`: *N*

Jumlah berapa kali handler `napi` memanggil `napi_complete` untuk antrean ini.

`queue_N_tx_tx_poll`: *N*

Jumlah berapa kali handler `napi` dijadwalkan untuk antrean ini.

`queue_N_tx_doorbells`: *N*

Jumlah doorbell transmisi untuk antrean ini.

`queue_N_tx_prepare_ctx_err`: *N*

Jumlah berapa kali `ena_com_prepare_tx` gagal untuk antrean ini.

`queue_N_tx_bad_req_id`: *N*

`req_id` tidak valid untuk antrean ini. `req_id` yang valid adalah nol, dikurangi `queue_size`, dikurangi 1.

`queue_N_tx_llq_buffer_copy`: *N*

Jumlah paket yang ukuran headernya lebih besar dari entri llq untuk antrean ini.

`queue_N_tx_missed_tx`: *N*

Jumlah paket yang tidak diselesaikan untuk antrean ini.

`queue_N_tx_unmask_interrupt`: *N*

Jumlah berapa kali interupsi tx dibuka untuk antrean ini.

`queue_N_rx_cnt`: *N*

Jumlah paket yang diterima untuk antrean ini.

`queue_N_rx_bytes`: *N*

Jumlah byte yang diterima untuk antrean ini.

`queue_N_rx_rx_copybreak_pkt`: *N*

Jumlah berapa kali antrean rx menerima paket yang kurang dari ukuran paket rx_copybreak untuk antrean ini.

`queue_N_rx_csum_good`: *N*

Jumlah berapa kali antrean rx menerima paket di mana checksum diperiksa dan benar untuk antrean ini.

`queue_N_rx_refil_partial`: *N*

Jumlah berapa kali driver tidak berhasil mengisi kembali bagian antrean rx yang kosong dengan buffer untuk antrean ini. Jika nilai ini bukan nol, ini menunjukkan sumber daya memori rendah.

`queue_N_rx_bad_csum`: *N*

Jumlah berapa kali antrean `rx` memiliki checksum buruk untuk antrean ini (hanya jika rx checksum offload didukung).

`queue_N_rx_page_alloc_fail`: *N*

Jumlah berapa kali alokasi halaman gagal untuk antrean ini. Jika nilai ini bukan nol, ini menunjukkan sumber daya memori rendah.

`queue_N_rx_skb_alloc_fail`: *N*

Jumlah berapa kali alokasi SKB gagal untuk antrean ini. Jika nilai ini bukan nol, ini menunjukkan sumber daya sistem yang rendah.

`queue_N_rx_dma_mapping_err`: *N*

Jumlah kesalahan akses memori langsung. Jika nilai ini bukan 0, ini menunjukkan sumber daya sistem yang rendah.

`queue_N_rx_bad_desc_num`: *N*

Terlalu banyak buffer per paket. Jika nilai ini bukan 0, ini menunjukkan penggunaan buffer yang sangat kecil.

`queue_N_rx_bad_req_id`: *N*

Req_id untuk antrean ini tidak valid. Req_id valid adalah dari [0, queue_size - 1].

`queue_N_rx_empty_rx_ring`: *N*

Jumlah berapa kali antrean rx kosong untuk antrean ini.

`queue_N_rx_csum_unchecked`: *N*

Jumlah berapa kali antrean rx menerima paket di mana checksum tidak diperiksa untuk antrean ini.

`queue_N_rx_xdp_aborted`: *N*

Jumlah berapa kali paket XDP diklasifikasikan sebagai XDP_ABORT.

`queue_N_rx_xdp_drop`: *N*

Jumlah berapa kali paket XDP diklasifikasikan sebagai XDP_DROP.

`queue_N_rx_xdp_pass`: *N*

Jumlah berapa kali paket XDP diklasifikasikan sebagai XDP_PASS.

`queue_N_rx_xdp_tx`: *N*

Jumlah berapa kali paket XDP diklasifikasikan sebagai XDP_TX.

`queue_N_rx_xdp_invalid`: *N*

Jumlah berapa kali kode pengembalian XDP untuk paket tidak valid.

`queue_N_rx_xdp_redirect`: *N*

Jumlah berapa kali paket XDP diklasifikasikan sebagai XDP_REDIRECT.

`queue_N_xdp_tx_cnt`: *N*

Jumlah paket yang dikirimkan untuk antrean ini.

`queue_N_xdp_tx_bytes`: *N*

Jumlah bita yang dikirimkan untuk antrean ini.

`queue_N_xdp_tx_queue_stop: N`

Jumlah berapa kali antrean ini penuh dan dihentikan.

`queue_N_xdp_tx_queue_wakeup: N`

Jumlah berapa kali antrean ini dilanjutkan setelah dihentikan.

`queue_N_xdp_tx_dma_mapping_err: N`

Jumlah kesalahan akses memori langsung. Jika nilai ini bukan 0, ini menunjukkan sumber daya sistem yang rendah.

`queue_N_xdp_tx_linearize: N`

Jumlah berapa kali linierisasi buffer XDP dicoba untuk antrean ini.

`queue_N_xdp_tx_linearize_failed: N`

Jumlah berapa kali linierisasi buffer XDP gagal untuk antrean ini.

`queue_N_xdp_tx_napi_comp: N`

Jumlah berapa kali handler napi memanggil napi_complete untuk antrean ini.

`queue_N_xdp_tx_tx_poll: N`

Jumlah berapa kali handler napi dijadwalkan untuk antrean ini.

`queue_N_xdp_tx_doorbells: N`

Jumlah doorbell transmisi untuk antrean ini.

`queue_N_xdp_tx_prepare_ctx_err: N`

Jumlah berapa kali ena_com_prepare_tx gagal untuk antrean ini. Nilai ini harus selalu nol; jika tidak, lihat log driver.

`queue_N_xdp_tx_bad_req_id: N`

Req_id untuk antrean ini tidak valid. Req_id valid adalah dari [0, queue_size - 1].

`queue_N_xdp_tx_llq_buffer_copy: N`

Jumlah paket yang headernya disalin menggunakan buffer llq untuk antrean ini.

`queue_N_xdp_tx_missed_tx: N`

Jumlah berapa kali entri antrean tx melewatkan waktu tunggu penyelesaian untuk antrean ini.

`queue_N_xdp_tx_unmask_interrupt: N`

Jumlah berapa kali interupsi tx dibuka untuk antrean ini.

`ena_admin_q_aborted_cmd: N`

Jumlah perintah admin yang dibatalkan. Ini biasanya terjadi selama prosedur pemulihan otomatis.

`ena_admin_q_submitted_cmd: N`

Jumlah doorbell antrean admin.

`ena_admin_q_completed_cmd: N`

Jumlah penyelesaian antrean admin.

`ena_admin_q_out_of_space: N`

Berapa kali driver mencoba mengirim perintah admin baru, namun antrean penuh.

`ena_admin_q_no_completion: N`

Berapa kali driver tidak mendapatkan penyelesaian perintah dari admin.

### Log error driver di syslog

Driver ENA menulis pesan log ke syslog selama boot sistem. Anda dapat memeriksa log ini untuk mencari error jika mengalami masalah. Di bawah ini adalah contoh informasi yang dicatat lognya oleh driver ENA di syslog selama boot sistem, bersama dengan beberapa penjelasan untuk pesan yang dipilih.

```
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  478.416939] [ENA_COM:
 ena_com_validate_version] ena device version: 0.10
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  478.420915] [ENA_COM:
 ena_com_validate_version] ena controller version: 0.0.1 implementation version 1
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.256831] ena 0000:00:03.0: Device
 watchdog is Enabled
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.672947] ena 0000:00:03.0: creating 8 io
 queues. queue size: 1024
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.680885] [ENA_COM:
 ena_com_init_interrupt_moderation] Feature 20 isn't supported  // Interrupt moderation
 is not supported by the device
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.691609] [ENA_COM:
 ena_com_get_feature_ex] Feature 10 isn't supported // RSS HASH function configuration
 is not supported by the
```

```
 device
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.694583] [ENA_COM:
 ena_com_get_feature_ex] Feature 18 isn't supported //RSS HASH input source
 configuration is not supported by the device
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.697433] [ENA_COM:
 ena_com_set_host_attributes] Set host attribute isn't supported
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.701064] ena 0000:00:03.0 (unnamed
 net_device) (uninitialized): Cannot set host attributes
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  479.704917] ena 0000:00:03.0: Elastic
 Network Adapter (ENA) found at mem f3000000, mac addr 02:8a:3c:1e:13:b5 Queues 8
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  480.805037] EXT4-fs (xvda1): re-mounted.
 Opts: (null)
Jun  3 22:37:46 ip-172-31-2-186 kernel: [  481.025842] NET: Registered protocol family
 10
```

Error mana yang dapat saya abaikan?

Peringatan berikut yang mungkin muncul di log error sistem Anda dapat diabaikan untuk Elastic Network Adapter:

Set atribut host tidak didukung

Atribut host tidak didukung untuk perangkat ini.

gagal untuk mengalokasikan buffer untuk antrean rx

Ini adalah error yang dapat dipulihkan, dan ini menunjukkan bahwa mungkin ada masalah tekanan memori saat error terjadi.

Fitur *X* tidak didukung

Fitur yang direferensikan tidak didukung oleh Elastic Network Adapter. Nilai yang mungkin untuk *X* meliputi:

- 10: Konfigurasi fungsi RSS Hash tidak didukung untuk perangkat ini.
- 12: Konfigurasi tabel Indirection RSS tidak didukung untuk perangkat ini.
- 18: Konfigurasi Input Hash RSS tidak didukung untuk perangkat ini.
- 20: Moderasi interupsi tidak didukung untuk perangkat ini.
- 27: Driver Elastic Network Adapter tidak mendukung polling kemampuan Ethernet dari snmpd.

Gagal untuk mengonfigurasi AENQ

Elastic Network Adapter tidak mendukung konfigurasi AENQ.

Mencoba untuk mengatur peristiwa AENQ yang tidak didukung

Error ini menunjukkan upaya untuk mengatur grup peristiwa AENQ yang tidak didukung oleh Elastic Network Adapter.

Sub-optimal pemberitahuan konfigurasi

Perangkat ENA mendeteksi pengaturan konfigurasi sub-optimal di driver yang dapat Anda ubah. Perangkat memberi tahu driver ENA dan mencatat peringatan ke konsol. Contoh berikut menunjukkan format pesan peringatan.

```
Sub-optimal configuration notification code: 1. Refer to AWS ENA documentation for
 additional details and mitigation options.
```

Daftar berikut menunjukkan detail kode notifikasi dan tindakan yang disarankan untuk temuan konfigurasi sub-optimal.

- Kode 1: ENA Express dengan konfigurasi LLQ lebar tidak disarankan

  ENA Ekspres ENI dikonfigurasi dengan LLQ lebar. Konfigurasi ini kurang optimal dan dapat memengaruhi performa ENA Ekspres. Kami menyarankan Anda menonaktifkan pengaturan LLQ yang luas saat Anda menggunakan ENA Ekspres ENIS sebagai berikut.

  ```
  sudo rmmod ena && sudo modprobe ena force_large_llq_header=0
  ```

  Untuk informasi selengkapnya tentang konfigurasi optimal untuk ENA Ekspres, lihat Meningkatkan kinerja jaringan antara instans EC2 dengan ENA Express.

- Kode 2: ENA Express ENI dengan kedalaman antrian Tx yang kurang optimal tidak disarankan

  ENA Ekspres ENI dikonfigurasi dengan kedalaman antrean Tx sub-optimal. Konfigurasi ini dapat memengaruhi performa ENA Ekspres. Kami menyarankan Anda memperbesar semua antrean Tx ke nilai maksimum untuk antarmuka jaringan saat Anda menggunakan ENA Ekspres ENIs sebagai berikut.

  Anda dapat menjalankan ethtool perintah berikut untuk menyesuaikan ukuran LLQ. Untuk mempelajari selengkapnya tentang cara mengontrol, menanyakan, dan mengaktifkan Wide-LLQ, lihat topik Large Qu Low-Latency eue (Large LLQ) dari dokumentasi driver kernel Linux untuk ENA di repositori Amazon Drivers. GitHub

```
ethtool -g interface
```

Atur antrian Tx Anda ke kedalaman maksimum:

```
ethtool -G interface tx depth
```

Untuk informasi selengkapnya tentang konfigurasi optimal untuk ENA Ekspres, lihat Meningkatkan kinerja jaringan antara instans EC2 dengan ENA Express.

- Kode 3: ENA dengan ukuran LLQ reguler dan lalu lintas paket Tx melebihi ukuran header maksimum yang didukung

  Secara default, ENA LLQ mendukung ukuran header paket Tx hingga 96 byte. Jika ukuran header paket lebih besar dari 96 byte, paket dijatuhkan. Untuk mengurangi masalah ini, kami sarankan Anda mengaktifkan Wide-LLQ, yang meningkatkan ukuran header paket Tx yang didukung hingga maksimum 224 byte.

  Namun, ketika Anda mengaktifkan Wide-LLQ, ukuran cincin Tx maksimum dikurangi dari 1000 menjadi 512 entri. Wide-LLQ diaktifkan secara default untuk semua jenis instance Nitro v4 dan yang lebih baru.

  - Jenis instance Nitro v4 memiliki ukuran cincin Wide-LLQ Tx maksimum default sebesar 512 entri, yang tidak dapat diubah.
  - Jenis instance Nitro v5 memiliki ukuran cincin Wide-LLQ Tx default sebesar 512 entri, yang dapat Anda tingkatkan hingga 1000 entri.

  Anda dapat menjalankan ethtool perintah berikut untuk menyesuaikan ukuran LLQ. Untuk mempelajari selengkapnya tentang cara mengontrol, menanyakan, dan mengaktifkan Wide-LLQ, lihat topik Large Qu Low-Latency eue (Large LLQ) dari dokumentasi driver kernel Linux untuk ENA di repositori Amazon Drivers. GitHub

  Temukan kedalaman maksimum untuk antrian Tx Anda:

  ```
  ethtool -g interface
  ```

  Atur antrian Tx Anda ke kedalaman maksimum:

  ```
  ethtool -G interface tx depth
  ```

# Memecahkan masalah driver Windows Adaptor Jaringan Elastic

Adaptor Jaringan Elastis (ENA) dirancang untuk meningkatkan kesehatan sistem operasi dan mengurangi perilaku atau kegagalan perangkat keras yang tidak terduga yang dapat mengganggu pengoperasian instans Windows Anda. Arsitektur ENA menjaga kegagalan perangkat atau driver setransparan mungkin ke sistem operasi.

## Kumpulkan informasi diagnostik pada instans

Langkah-langkah untuk membuka alat sistem operasi (OS) Windows bervariasi, tergantung pada versi OS yang diinstal pada instans Anda. Di bagian berikut, kami menggunakan dialog Run untuk membuka alat, yang bekerja sama di semua versi OS. Namun, Anda dapat mengakses alat ini menggunakan metode apa pun yang Anda inginkan.

### Akses dialog Jalankan

- Menggunakan kombinasi tombol logo Windows: `Windows` + `R`
- Menggunakan bilah pencarian:
  - Masukkan `run` ke bilah pencarian.
  - Pilih aplikasi Jalankan dari hasil pencarian.

Beberapa langkah memerlukan menu konteks untuk mengakses properti atau tindakan peka konteks. Ada beberapa cara untuk melakukan ini, tergantung pada versi OS dan perangkat keras Anda.

### Akses menu konteks

- Menggunakan mouse Anda: klik kanan item untuk membuka menu konteksnya.
- Menggunakan keyboard Anda:
  - Tergantung pada versi OS Anda, gunakan `Shift` + `F10`, atau `Ctrl` + `Shift` + `F10`.
  - Jika Anda memiliki tombol konteks pada keyboard Anda (tiga garis horizontal dalam kotak), pilih item yang Anda inginkan dan kemudian tekan tombol konteks.

Jika Anda dapat terhubung ke instans Anda, gunakan teknik berikut untuk mengumpulkan informasi diagnostik untuk pemecahan masalah.

### Periksa status perangkat ENA

Untuk memeriksa status driver ENA Windows Anda menggunakan Windows Device Manager, ikuti langkah-langkah berikut:

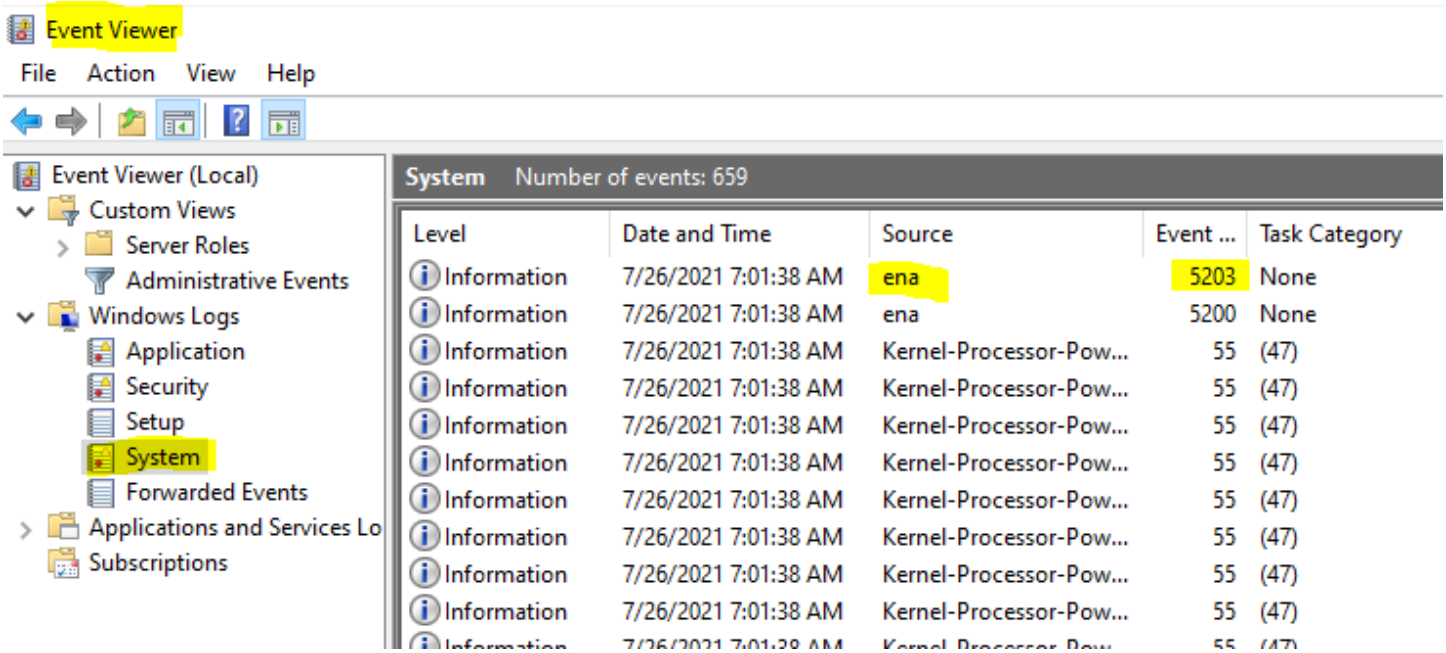
1. Buka dialog Jalankan menggunakan salah satu metode yang dijelaskan di bagian sebelumnya.
2. Untuk membuka Windows Device Manager, masukkan `devmgmt.msc` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Device Manager.
4. Pilih panah di sebelah kiri Adaptor jaringan untuk memperluas daftar.
5. Pilih nama, atau buka menu konteks untuk Adaptor Jaringan Elastis Amazon, lalu pilih Properti. Ini membuka dialog Properti Adaptor Jaringan Elastis Amazon.
6. Verifikasi bahwa pesan di tab Umum mengatakan “Perangkat ini berfungsi dengan baik”.

### Selidiki pesan peristiwa driver

Untuk meninjau log peristiwa driver ENA Windows menggunakan Windows Event Viewer, ikuti langkah-langkah berikut:

1. Buka dialog Jalankan menggunakan salah satu metode yang dijelaskan di bagian sebelumnya.
2. Untuk membuka Windows Event Viewer, masukkan `eventvwr.msc` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Event Viewer.
4. Perluas menu Windows Logs, lalu pilih System.
5. Di bawah Tindakan, di panel kanan atas, pilih Filter Log Saat Ini. Ini menampilkan dialog penyaringan.
6. Di kotak Sumber peristiwa, masukkan `ena`. Ini membatasi hasil untuk peristiwa yang dihasilkan oleh driver ENA Windows.
7. Pilih OKE. Ini menunjukkan hasil log peristiwa yang difilter di bagian detail jendela.
8. Untuk menelusuri detailnya, pilih pesan peristiwa dari daftar.

Contoh berikut menunjukkan peristiwa driver ENA dalam daftar peristiwa sistem Windows Event Viewer:




Ringkasan pesan peristiwa

Tabel berikut menunjukkan pesan peristiwa yang dihasilkan oleh driver ENA Windows.

Input

| ID peristiwa | Deskripsi peristiwa driver ENA | Tipe |
|---|---|---|
| 5001 | Perangkat keras kehabisan sumber daya | Kesalahan |
| 5002 | Adaptor telah mendeteksi kesalahan perangkat keras | Kesalahan |
| 5005 | Adaptor telah habis waktu operasi NDIS yang tidak selesai tepat waktu | Kesalahan |
| 5032 | Adaptor gagal mengatur ulang perangkat | Kesalahan |
| 5200 | Adaptor telah diinisialisasi | Informasi |
| 5201 | Adaptor telah dihentikan | Informasi |

| ID peristiwa | Deskripsi peristiwa driver ENA | Tipe |
|---|---|---|
| 5202 | Adaptor telah dijeda | Informasi |
| 5203 | Adaptor telah dimulai ulang | Informasi |
| 5204 | Adaptor telah dimatikan | Informasi |
| 5205 | Adaptor telah diatur ulang | Kesalahan |
| 5206 | Adaptor telah dihapus secara mengejutkan | Kesalahan |
| 5208 | Rutin inisialisasi adaptor telah gagal | Kesalahan |
| 5210 | Adaptor telah mengalami dan berhasil memulihkan masalah internal | Kesalahan |

Tinjau metrik performa

Driver ENA Windows menerbitkan metrik performa jaringan dari instans di mana metrik diaktifkan. Anda dapat melihat dan mengaktifkan metrik pada instans menggunakan aplikasi Monitor Performa asli. Untuk informasi selengkapnya tentang metrik yang diproduksi driver ENA Windows, lihat Memantau kinerja jaringan untuk pengaturan ENA pada instans EC2 Anda.

Pada contoh di mana metrik ENA diaktifkan, dan CloudWatch agen Amazon diinstal, CloudWatch mengumpulkan metrik yang terkait dengan penghitung di Monitor Kinerja Windows, serta beberapa metrik lanjutan untuk ENA. Metrik ini dikumpulkan sebagai tambahan untuk metrik yang diaktifkan secara default di instans EC2. Untuk informasi selengkapnya tentang metrik, lihat Metrik yang dikumpulkan oleh CloudWatch agen di Panduan CloudWatch Pengguna Amazon.

> **Note**
>
> Metrik performa tersedia untuk driver ENA versi 2.4.0 dan yang lebih baru (juga untuk versi 2.2.3). Driver ENA versi 2.2.4 dibatalkan karena potensi penurunan performa pada instans

EC2 generasi keenam. Kami menyarankan Anda melakukan peningkatan ke versi driver saat ini untuk memastikan bahwa Anda memiliki pembaruan terbaru.

Beberapa cara yang dapat Anda pakai untuk menggunakan metrik performa meliputi:

- Memecahkan masalah performa instans.
- Pilih ukuran instans yang tepat untuk beban kerja.
- Merencanakan kegiatan penskalaan secara proaktif.
- Benchmark aplikasi untuk menentukan apakah mereka memaksimalkan performa yang tersedia pada sebuah instans.

### Tingkat penyegaran

Secara default, driver menyegarkan metrik menggunakan interval 1 detik. Namun, aplikasi yang mengambil metrik mungkin menggunakan interval yang berbeda untuk polling. Anda dapat mengubah interval penyegaran di Device Manager, menggunakan properti lanjutan untuk driver.

Untuk mengubah interval penyegaran metrik untuk driver ENA Windows, ikuti langkah-langkah berikut:

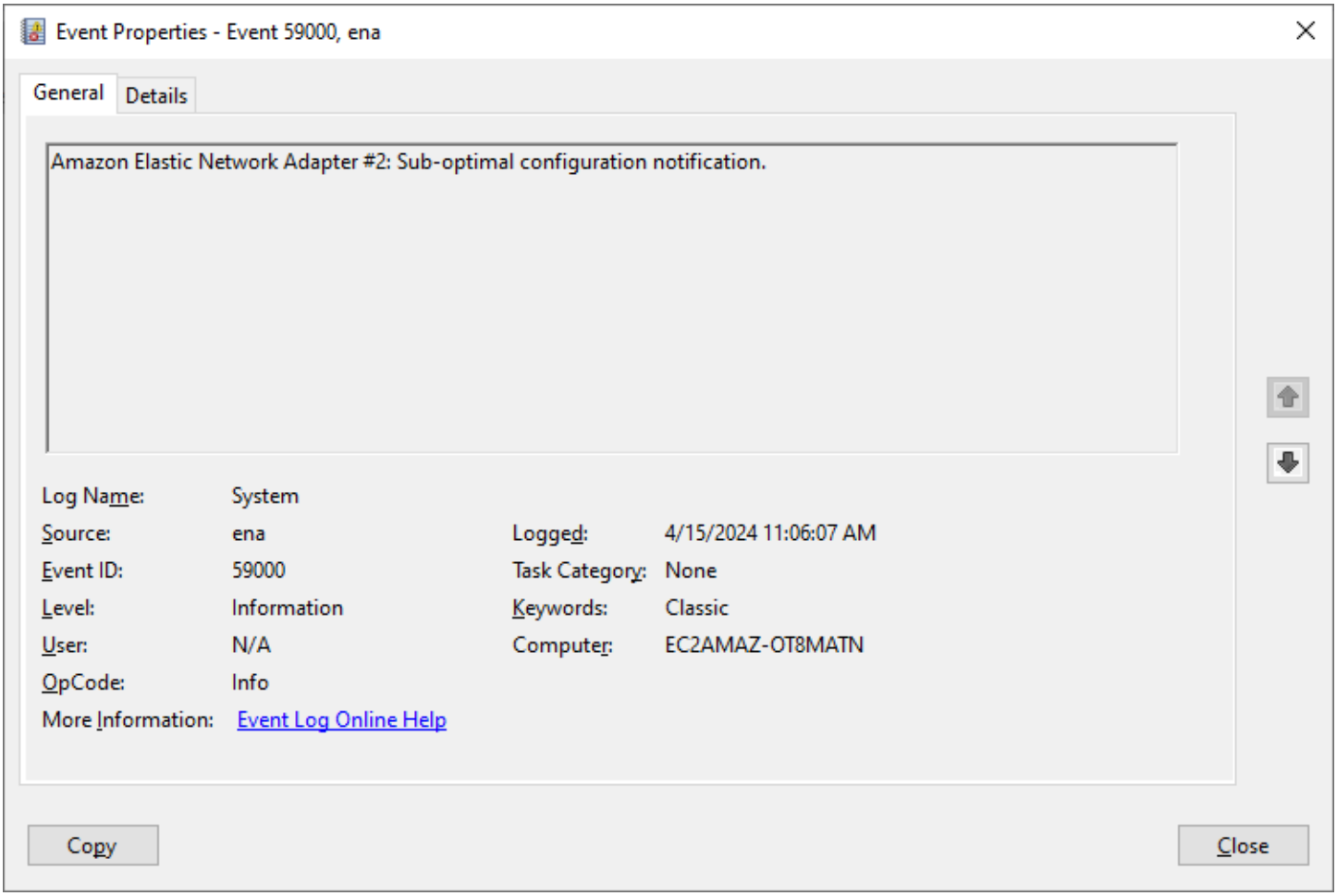
1. Buka dialog Jalankan menggunakan salah satu metode yang dijelaskan di bagian sebelumnya.
2. Untuk membuka Windows Device Manager, masukkan `devmgmt.msc` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Device Manager.
4. Pilih panah di sebelah kiri Adaptor jaringan untuk memperluas daftar.
5. Pilih nama, atau buka menu konteks untuk Adaptor Jaringan Elastis Amazon, lalu pilih Properti. Ini membuka dialog Properti Adaptor Jaringan Elastis Amazon.
6. Buka tab Advanced di jendela pop-up.
7. Dari daftar Properti, pilih Metrics Refresh Interval untuk mengubah nilai.
8. Setelah selesai, pilih OK.

### Selidiki pemberitahuan konfigurasi yang kurang optimal

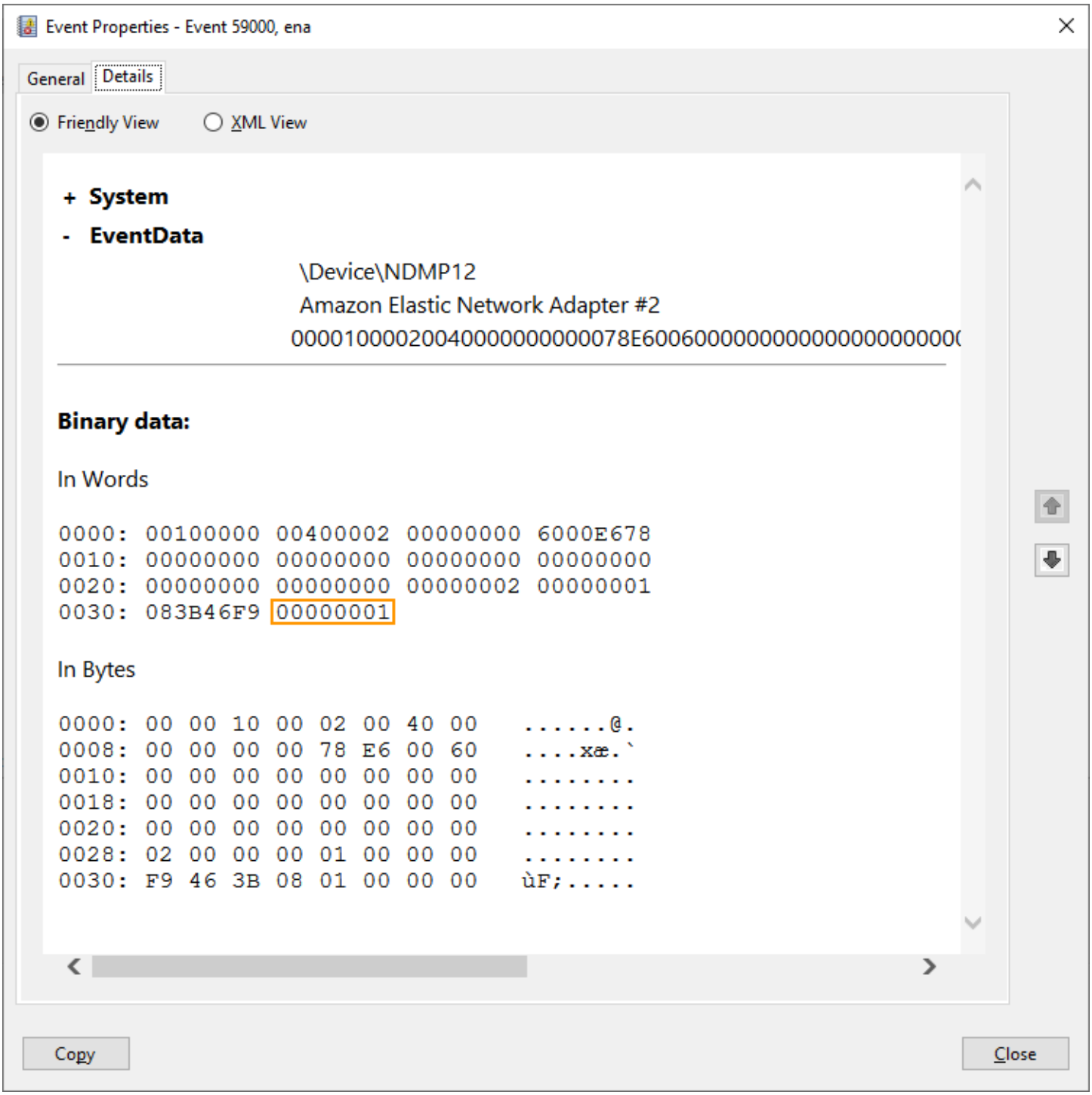
Perangkat ENA mendeteksi pengaturan konfigurasi sub-optimal di driver yang dapat Anda ubah. Perangkat memberi tahu driver ENA dan mencatat pemberitahuan peristiwa. Untuk meninjau peristiwa yang kurang optimal di Windows Event Viewer

1. Buka dialog Jalankan menggunakan salah satu metode yang dijelaskan di bagian sebelumnya.
2. Untuk membuka Windows Event Viewer, masukkan `eventvwr.msc` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Event Viewer.
4. Perluas menu Windows Logs, lalu pilih System.
5. Di bawah Tindakan, di panel kanan atas, pilih Filter Log Saat Ini. Ini menampilkan dialog penyaringan.
6. Di kotak Sumber peristiwa, masukkan `ena`. Ini membatasi hasil untuk peristiwa yang dihasilkan oleh driver ENA Windows.
7. Pilih OKE. Ini menunjukkan hasil log peristiwa yang difilter di bagian detail jendela.

Peristiwa dengan ID `59000` memberi tahu Anda tentang temuan konfigurasi yang kurang optimal. Buka menu konteks (klik kanan) untuk acara dan pilih Properti Acara untuk membuka tampilan terperinci, atau pilih Panel Pr atinjau dari menu Lihat untuk melihat detail yang sama.



Buka tab Detail untuk melihat kode acara. Di bagian Data Biner: Dalam kata-kata, kata terakhir adalah kode.




Daftar berikut menunjukkan detail kode notifikasi dan tindakan yang disarankan untuk temuan konfigurasi sub-optimal.

- Kode**1**: ENA Ekspres dengan konfigurasi LLQ lebar tidak disarankan

ENA Ekspres ENI dikonfigurasi dengan LLQ lebar. Konfigurasi ini kurang optimal dan dapat memengaruhi performa ENA Ekspres. Kami menyarankan Anda menonaktifkan pengaturan LLQ yang luas saat Anda menggunakan ENA Ekspres ENIS sebagai berikut.

1. Untuk membuka Windows Device Manager, masukkan `devmgmt.msc` di kotak Jalankan.
2. Pilih OKE. Ini membuka jendela Device Manager.
3. Pilih panah di sebelah kiri Adaptor jaringan untuk memperluas daftar.
4. Buka properti perangkat untuk`Amazon Elastic Network Adapter`.
5. Dari sana, buka tab Advanced untuk membuat perubahan.
6. Pilih properti Kebijakan Ukuran Header LLQ, dan atur nilainya ke`Normal (128 Bytes)`.
7. Pilih OK untuk menyimpan perubahan Anda.

- Kode**2**: ENA Ekspres ENI dengan kedalaman antrean Tx sub-optimal tidak disarankan

  ENA Ekspres ENI dikonfigurasi dengan kedalaman antrean Tx sub-optimal. Konfigurasi ini dapat memengaruhi performa ENA Ekspres. Kami menyarankan Anda memperbesar semua antrean Tx ke nilai maksimum untuk antarmuka jaringan saat Anda menggunakan ENA Ekspres ENIs sebagai berikut.

  Ikuti langkah-langkah ini untuk memperbesar antrian Tx ke kedalaman maksimum:

  1. Untuk membuka Windows Device Manager, masukkan `devmgmt.msc` di kotak Jalankan.
  2. Pilih OKE. Ini membuka jendela Device Manager.
  3. Pilih panah di sebelah kiri Adaptor jaringan untuk memperluas daftar.
  4. Buka properti perangkat untuk`Amazon Elastic Network Adapter`.
  5. Dari sana, buka tab Advanced untuk membuat perubahan.
  6. Pilih properti Transmit Buffers, dan atur nilainya ke maksimum yang didukung.
  7. Pilih OK untuk menyimpan perubahan Anda.

### Reset adaptor ENA

Proses reset dimulai ketika driver ENA Windows mendeteksi kesalahan pada adaptor, dan menandai adaptor sebagai tidak sehat. Driver tidak dapat mengatur ulang sendiri, jadi itu tergantung pada sistem operasi untuk memeriksa status kesehatan adaptor, dan memanggil pegangan reset untuk driver ENA Windows. Proses reset dapat mengakibatkan periode waktu singkat di mana kehilangan lalu lintas terjadi. Namun, koneksi TCP harus dapat pulih.

Adaptor ENA mungkin juga secara tidak langsung meminta prosedur reset perangkat, dengan gagal mengirim notifikasi keep-alive. Misalnya, jika adaptor ENA mencapai status tidak diketahui setelah memuat konfigurasi yang tidak dapat dipulihkan, mungkin berhenti mengirim notifikasi keep-alive.

Penyebab umum reset untuk adaptor ENA

- Keep-alive pesan hilang

  Adaptor ENA memposting peristiwa keep-alive dengan kecepatan tetap (biasanya sekali setiap detik). Driver ENA Windows mengimplementasikan mekanisme watchdog, yang secara berkala memeriksa keberadaan pesan keep-alive ini. Jika mendeteksi satu atau lebih pesan baru sejak terakhir kali diperiksa, itu mencatat hasil yang sukses. Jika tidak, pengemudi menyimpulkan bahwa perangkat mengalami kegagalan, dan memulai urutan reset.
- Paket terjebak dalam antrean transmisi

  Adaptor ENA memverifikasi bahwa paket mengalir melalui antrean transmisi seperti yang diharapkan. Driver ENA Windows mendeteksi jika paket macet, dan memulai urutan reset jika ada.
- Batas waktu baca untuk register Memory Mapped I/O (MMIO)

  Untuk membatasi operasi pembacaan memori I/O yang dipetakan (MMIO), driver ENA Windows mengakses register MMIO hanya selama proses inisialisasi dan reset. Jika driver mendeteksi batas waktu, dibutuhkan salah satu tindakan berikut, tergantung pada proses apa yang sedang berjalan:
  - Jika batas waktu terdeteksi selama inisialisasi, aliran gagal, yang mengakibatkan driver menampilkan tanda seru kuning oleh adaptor ENA di Windows Device Manager.
  - Jika batas waktu terdeteksi selama reset, alirannya gagal. OS kemudian memulai penghapusan kejutan adaptor ENA, dan memulihkannya dengan menghentikan dan memulai adaptor yang telah dihapus. Untuk informasi selengkapnya tentang penghapusan kejutan kartu antarmuka jaringan (NIC), lihat Menangani Penghapusan Kejutan NIC dalam dokumentasi Developer Perangkat Keras Microsoft Windows.

Skenario pemecahan masalah

Skenario berikut dapat membantu Anda memecahkan masalah yang mungkin Anda alami dengan driver ENA Windows. Kami menyarankan Anda memulai dengan meningkatkan driver ENA Anda, jika Anda tidak memiliki versi terbaru. Untuk menemukan driver terbaru untuk versi OS Windows Anda, lihat Lacak rilis versi driver ENA Windows.

Versi driver ENA yang tidak terduga diinstal

Deskripsi

Setelah Anda melalui langkah-langkah untuk menginstal versi tertentu dari driver ENA, Windows Device Manager menunjukkan bahwa Windows menginstal versi driver ENA yang berbeda.

Penyebab

Ketika Anda menjalankan instalasi untuk paket driver, Windows memberi peringkat semua paket driver yang valid untuk perangkat yang diberikan di Toko Driver lokal sebelum dimulai. Kemudian memilih paket dengan nilai peringkat terendah sebagai kecocokan terbaik. Ini bisa berbeda dari paket yang ingin Anda instal. Untuk informasi selengkapnya tentang proses pemilihan paket driver perangkat, lihat Cara Windows memilih paket driver untuk perangkat di situs web dokumentasi Microsoft.

Solusi

Untuk memastikan bahwa Windows menginstal versi paket driver yang Anda pilih, Anda dapat menghapus paket driver berperingkat lebih rendah dari Toko Driver dengan alat baris perintah PnPUtil.

Ikuti langkah-langkah berikut ini untuk memperbarui driver ENA:

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
2. Buka jendela properti Device Manager, seperti yang dijelaskan di Periksa status perangkat ENA bagian. Ini membuka tab Umum jendela Properti Adaptor Jaringan Elastis Amazon.
3. Buka tab Driver.
4. Pilih Perbarui Driver. Ini membuka kotak dialog Perbarui Driver Perangkat lunak – Adaptor Jaringan Elastis Amazon.
    a. Pada Bagaimana Anda ingin mencari perangkat lunak driver? halaman, pilih Jelajahi komputer saya untuk perangkat lunak driver.
    b. Pada halaman Jelajahi driver perangkat lunak di komputer Anda, pilih Biarkan saya memilih dari daftar driver perangkat di komputer saya, yang terletak di bawah bilah pencarian.
    c. Pada Pilih driver perangkat yang ingin Anda instal untuk halaman perangkat keras ini, pilih Have Disk....
    d. Di jendela Instal dari Disk, pilih Browse..., di sebelah lokasi file dari daftar dropdown.

e. Arahkan ke lokasi di mana Anda mengunduh paket driver ENA target. Pilih file bernama `ena.inf` dan pilih Buka.

f. Untuk memulai instalasi, pilih OK, lalu pilih Selanjutnnya.

5. Jika penginstal tidak secara otomatis me-reboot instance Anda, jalankan Restart-Computer PowerShell cmdlet.

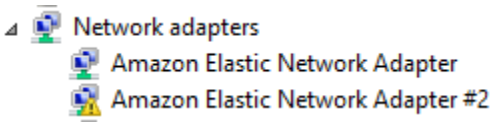
```
PS C:\> Restart-Computer
```

Peringatan perangkat untuk driver ENA

Deskripsi

Ikon adaptor ENA di bagian Adaptor Jaringan Manajer Perangkat menampilkan tanda peringatan (segitiga kuning dengan tanda seru di dalamnya).

Contoh berikut menunjukkan adaptor ENA dengan ikon peringatan di Windows Device Manager:




Penyebab

Peringatan perangkat ini umumnya disebabkan oleh masalah lingkungan, yang mungkin memerlukan lebih banyak penelitian, dan seringkali memerlukan proses eliminasi untuk menentukan penyebab yang mendasarinya. Untuk daftar lengkap kesalahan perangkat, lihat Pesan Kesalahan Pengelola Perangkat di dokumentasi Microsoft.

Solusi

Solusi untuk peringatan perangkat ini tergantung pada akar penyebabnya. Proses eliminasi yang dijelaskan di sini mencakup beberapa langkah dasar untuk membantu mengidentifikasi dan menyelesaikan masalah paling umum yang mungkin memiliki solusi sederhana. Analisis akar penyebab tambahan diperlukan ketika langkah-langkah ini tidak menyelesaikan masalah.

Ikuti langkah-langkah berikut untuk membantu mengidentifikasi dan menyelesaikan masalah umum:

1. Hentikan dan mulai perangkat

   Buka jendela properti Device Manager, seperti yang dijelaskan di Periksa status perangkat ENA bagian. Ini membuka tab Umum jendela Properti Adaptor Jaringan Elastis Amazon, di mana status Perangkat menampilkan kode kesalahan dan pesan singkat.

   a. Buka tab Driver.
   b. Pilih Nonaktifkan Perangkat, dan tanggapi Ya pada pesan peringatan yang ditampilkan.
   c. Pilih Aktifkan Perangkat.

2. Hentikan dan mulai instans EC2

   Jika adaptor masih menampilkan ikon peringatan di Device Manager, langkah selanjutnya adalah menghentikan dan memulai instans EC2. Ini meluncurkan kembali instans pada perangkat keras yang berbeda dalam banyak kasus.

3. Selidiki kemungkinan masalah sumber daya instans

   Jika Anda telah menghentikan dan memulai instans EC2 Anda, dan masalah tetap ada, ini mungkin menunjukkan masalah sumber daya pada instans Anda, seperti memori yang tidak mencukupi.

Batas waktu koneksi dengan reset adaptor (kode kesalahan 5007, 5205)

Deskripsi

Windows Event Viewer menunjukkan batas waktu adaptor dan mengatur ulang peristiwa yang terjadi dalam kombinasi untuk adaptor ENA. Pesan menyerupai contoh berikut:

- ID Peristiwa 5007: Adaptor Jaringan Elastis Amazon: Habis waktu selama operasi.
- ID Peristiwa 5205: Adaptor Jaringan Elastis Amazon: Atur ulang adaptor telah dimulai.

Atur ulang adaptor menyebabkan gangguan lalu lintas minimal. Bahkan ketika ada beberapa reset, itu tidak biasa bagi mereka untuk menyebabkan gangguan jaringan yang parah.

Penyebab

Urutan peristiwa ini menunjukkan bahwa driver ENA Windows memulai reset untuk adaptor ENA yang tidak responsif. Namun, mekanisme yang digunakan driver perangkat untuk mendeteksi masalah ini tunduk pada positif palsu akibat kelaparan CPU 0.

Solusi

Jika kombinasi kesalahan ini sering terjadi, periksa alokasi sumber daya Anda untuk melihat di mana penyesuaian mungkin bermanfaat.

1. Buka dialog Jalankan menggunakan salah satu metode yang dijelaskan di bagian sebelumnya.
2. Untuk membuka Windows Resource Monitor, masukkan `resmon` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Resource Monitor.
4. Buka tab CPU. Per-CPU grafik penggunaan ditampilkan di sepanjang sisi kanan jendela Resource Monitor.
5. Periksa tingkat penggunaan untuk CPU 0 untuk melihat apakah mereka terlalu tinggi.

Kami menyarankan Anda mengonfigurasi RSS untuk mengecualikan CPU 0 untuk adaptor ENA pada tipe instans yang lebih besar (lebih dari 16 vCPU). Untuk tipe instans yang lebih kecil, mengonfigurasi RSS dapat meningkatkan pengalaman, tetapi karena jumlah inti yang tersedia lebih rendah, pengujian diperlukan untuk memastikan bahwa membatasi inti CPU tidak berdampak negatif pada performa.

Gunakan perintah Set-NetAdapterRss untuk mengonfigurasi RSS untuk adaptor ENA Anda, seperti yang ditunjukkan dalam contoh berikut ini.

```
Set-NetAdapterRss -name (Get-NetAdapter | Where-Object {$_.InterfaceDescription -like
 "*Elastic*"}).Name -Baseprocessorgroup 0 -BaseProcessorNumber 1
```

Migrasi ke infrastruktur instans generasi keenam berdampak pada performa atau keterikatan

Deskripsi

Jika Anda bermigrasi ke instans EC2 generasi keenam, Anda mungkin mengalami penurunan performa atau kegagalan lampiran ENA jika Anda belum memperbarui versi driver ENA Windows Anda.

Penyebab

Tipe instans EC2 generasi keenam memerlukan versi minimum driver ENA Windows berikut, berdasarkan sistem operasi instans (OS).

Versi minimum

| Versi Windows Server | Versi driver ENA |
| --- | --- |
| Windows Server 2008 R2 | 2.2.3 atau 2.4.0 |
| Windows Server 2012 dan yang lebih baru | 2.2.3 dan versi yang lebih baru |
| Stasiun Kerja Windows | 2.2.3 dan versi yang lebih baru |

Solusi

Sebelum Anda meningkatkan ke instans EC2 generasi keenam, pastikan AMI yang Anda luncurkan memiliki driver yang kompatibel berdasarkan OS instans seperti yang ditunjukkan pada tabel sebelumnya. Untuk informasi selengkapnya, lihat Apa yang harus saya lakukan sebelum memigrasikan instans EC2 saya ke instans generasi keenam untuk memastikan bahwa saya mendapatkan performa jaringan yang maksimal? di pusat AWS re:Post pengetahuan.

Performa suboptimal untuk antarmuka jaringan elastis

Deskripsi

Antarmuka ENA tidak berfungsi seperti yang diharapkan.

Penyebab

Analisis akar penyebab untuk masalah performa adalah proses eliminasi. Ada terlalu banyak variabel yang terlibat untuk menyebutkan penyebab umum.

Solusi

Langkah pertama dalam analisis akar penyebab Anda adalah meninjau informasi diagnostik untuk instans yang tidak berfungsi seperti yang diharapkan, untuk menentukan apakah ada kesalahan yang mungkin menyebabkan masalah. Untuk informasi selengkapnya, lihat bagian Kumpulkan informasi diagnostik pada instans.

Anda mungkin perlu memodifikasi sistem operasi default untuk mencapai performa jaringan maksimum pada instans dengan jaringan yang ditingkatkan. Beberapa optimasi, seperti mengaktifkan

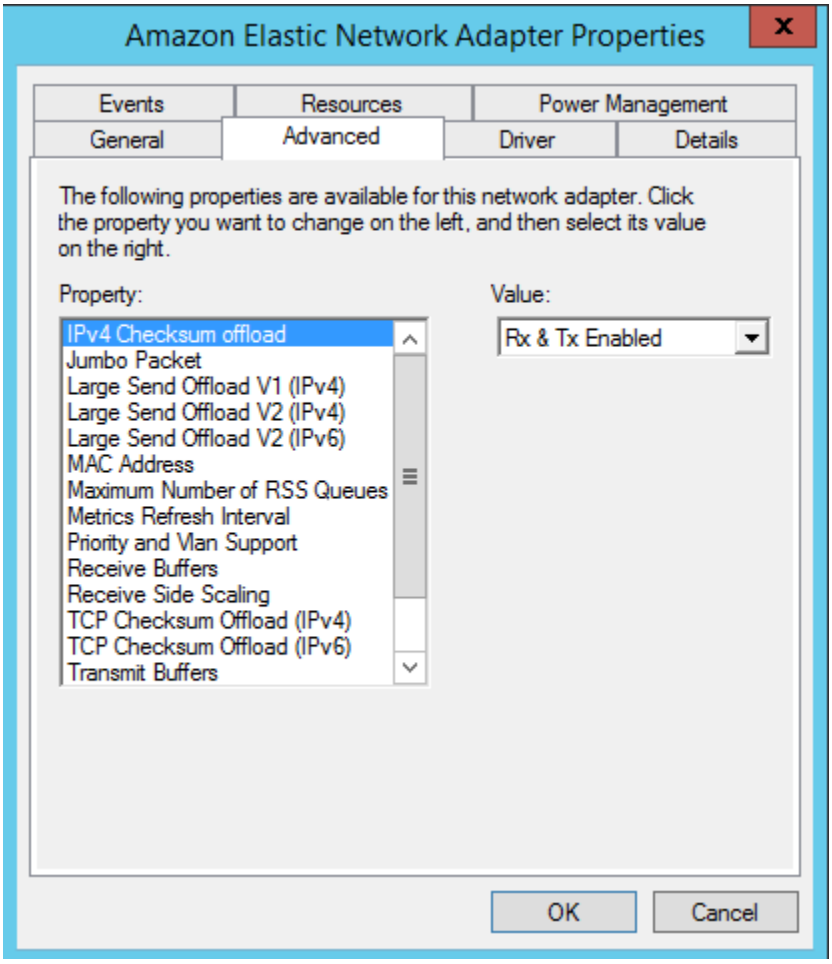
checksum offloading dan mengaktifkan RSS, dikonfigurasi secara default di AMI Windows resmi. Untuk pengoptimalan lain yang dapat Anda terapkan ke adaptor ENA, lihat penyesuaian performa yang ditunjukkan di Penyesuaian performa adaptor ENA.

Sebaiknya lanjutkan dengan hati-hati, dan batasi penyesuaian properti perangkat ke yang tercantum di bagian ini, atau perubahan tertentu yang direkomendasikan oleh tim AWS dukungan.

Untuk mengubah properti adaptor ENA, ikuti langkah-langkah ini:

1. Buka dialog Jalankan menggunakan salah satu metode yang dijelaskan di bagian sebelumnya.
2. Untuk membuka Windows Device Manager, masukkan `devmgmt.msc` di kotak Jalankan.
3. Pilih OKE. Ini membuka jendela Device Manager.
4. Pilih panah di sebelah kiri Adaptor jaringan untuk memperluas daftar.
5. Pilih nama, atau buka menu konteks untuk Adaptor Jaringan Elastis Amazon, lalu pilih Properti. Ini membuka dialog Properti Adaptor Jaringan Elastis Amazon.
6. Untuk membuat perubahan, buka tab Advanced.
7. Setelah selesai, pilih OKE untuk menyimpan perubahan Anda.

Contoh berikut menunjukkan properti adaptor ENA di Windows Device Manager:




Penyesuaian performa adaptor ENA

Tabel berikut mencakup properti yang dapat disesuaikan untuk meningkatkan performa antarmuka ENA.

Input

| Properti | Deskripsi | Nilai default | Penyesuaian |
|---|---|---|---|
| Menerima Buffer | Mengontrol jumlah entri dalam perangkat lunak menerima antrean. | 1024 | Dapat ditingkatkan hingga maksimum 8192. |
| Terima Penskalaan Samping (RSS) | Memungkinkan distribusi pemrosesa | Enabled | Anda dapat menyebarkan beban |

| Properti | Deskripsi | Nilai default | Penyesuaian |
| --- | --- | --- | --- |
| | n penerimaan jaringan yang efisien di beberapa CPU dalam sistem multiprosesor. | | di beberapa prosesor. Untuk mempelaja ri selengkapnya, lihat Mengoptim alkan kinerja jaringan pada instans EC2 Windows. |

| Properti | Deskripsi | Nilai default | Penyesuaian |
| --- | --- | --- | --- |
| Jumlah Antrean RSS Maksimum | Mengatur jumlah maksimum antrean RSS yang diizinkan saat RSS diaktifkan. | 32 | Jumlah antrean RSS ditentukan selama inisialisasi pengemudi, dan mencakup batasan berikut (antara lain):<br>• Batas antrean RSS ditetapkan oleh properti ini<br>• Batas instans (jumlah vCPU)<br>• Batas pembuatan perangkat keras (hingga 8 antrian RSS di eNAV1, dan hingga 32 antrean RSS di eNAV2)<br><br>Anda dapat mengatur nilai dari 1-32, tergantung pada instans dan batas pembuatan perangkat keras Anda. Untuk mempelajari selengkapnya, lihat Mengoptimalkan kinerja jaringan |

| Properti | Deskripsi | Nilai default | Penyesuaian |
| --- | --- | --- | --- |
| | | | pada instans EC2 Windows. |
| Paket jumbo | Memungkinkan penggunaan bingkai ethernet jumbo (lebih dari 1500 byte muatan). | Dinonaktifkan (ini membatasi muatan hingga 1500 byte atau kurang) | Nilai dapat diatur ke 9015, yang diterjemahkan ke 9001 byte payload. Ini adalah muatan maksimum untuk bingkai ethernet jumbo. Lihat Pertimbangan untuk menggunakan frame ethernet jumbo. |

Pertimbangan untuk menggunakan frame ethernet jumbo

Bingkai jumbo memungkinkan lebih dari 1500 byte data dengan meningkatkan ukuran payload per paket, yang meningkatkan persentase paket yang bukan overhead paket. Diperlukan lebih sedikit paket untuk mengirimkan data yang dapat digunakan dalam jumlah sama. Namun, lalu lintas dibatasi hingga MTU maksimum 1500 dalam kasus berikut:

- Lalu lintas di luar Wil AWS ayah tertentu untuk EC2 Classic.
- Lalu lintas di luar VPC tunggal.
- Lalu lintas melalui koneksi peering VPC antar wilayah.
- Lalu lintas melalui koneksi VPN.
- Lalu lintas melalui gateway internet.

> **Note**
>
> Paket lebih dari 1500 byte terfragmentasi. Jika Anda memiliki `Don't Fragment` bendera yang disetel di header IP, paket-paket ini dijatuhkan.
> Frame Jumbo harus digunakan dengan hati-hati untuk lalu lintas internet-bound atau setiap lalu lintas yang meninggalkan VPC. Paket difragmentasi oleh sistem menengah, yang

memperlambat lalu lintas ini. Untuk menggunakan bingkai jumbo di dalam VPC tanpa memengaruhi lalu lintas keluar yang meninggalkan VPC, coba salah satu opsi berikut:

- Konfigurasikan ukuran MTU berdasarkan rute.
- Gunakan beberapa antarmuka jaringan dengan ukuran MTU yang berbeda dan rute yang berbeda.

Kasus penggunaan yang disarankan untuk bingkai jumbo

Bingkai jumbo dapat berguna untuk lalu lintas di dalam dan di antara VPC. Kami merekomendasikan penggunaan bingkai jumbo untuk kasus penggunaan berikut:

- Untuk instans yang ditempatkan di dalam grup penempatan klaster, bingkai jumbo membantu mencapai throughput jaringan semaksimum mungkin. Untuk informasi selengkapnya, lihat Grup penempatan untuk instans Amazon EC2 Anda.
- Anda dapat menggunakan frame jumbo untuk lalu lintas antara VPC dan jaringan on-premise Anda melalui Direct Connect. Untuk informasi selengkapnya tentang menggunakan Direct Connect, dan memverifikasi kemampuan bingkai jumbo, lihat MTU untuk antarmuka virtual pribadi atau antarmuka virtual transit di Panduan Direct Connect Pengguna.
- Untuk informasi selengkapnya tentang ukuran MTU yang didukung untuk gateway transit, lihat Kuota untuk gateway transit Anda di Panduan Pengguna Gateway Transit Amazon VPC.

## Meningkatkan kinerja jaringan antara instans EC2 dengan ENA Express

ENA Express didukung oleh teknologi AWS Scalable Reliable Datagram (SRD). SRD adalah protokol transportasi jaringan performa tinggi yang menggunakan perutean dinamis untuk meningkatkan throughput dan meminimalkan latensi ekor. Dengan ENA Express, Anda dapat berkomunikasi antara dua instans EC2 di Zona Ketersediaan yang sama atau di Zona Ketersediaan dalam Wilayah yang sama.

Manfaat ENA Ekspres

- Meningkatkan bandwidth maksimum yang dapat digunakan aliran tunggal dari 5 Gbps hingga 25 Gbps dalam Wilayah yang sama, hingga batas instans agregat.
- Mengurangi latensi ekor lalu lintas jaringan antara instans EC2 di Zona Ketersediaan yang sama, terutama selama periode beban jaringan tinggi.

- Mendeteksi dan menghindari jalur jaringan yang padat.
- Menangani beberapa tugas secara langsung di lapisan jaringan, seperti penataan ulang paket di ujung penerima, dan sebagian besar transmisi ulang yang diperlukan. Ini membebaskan lapisan aplikasi untuk pekerjaan lain.

> **Note**
>
> - Jika aplikasi Anda memiliki persyaratan paket per detik yang tinggi dan perlu mengoptimalkan latensi selama periode tidak padat, mungkin lebih cocok. Jaringan yang ditingkatkan
> - Lalu lintas ENA Express tidak dapat dikirim di Zona Lokal.
> - Dukungan ENA Express untuk lalu lintas antara Zona Ketersediaan tidak tersedia di Amerika Selatan (São Paulo), Timur Tengah (Bahrain), dan Timur Tengah (UEA).

Setelah Anda mengaktifkan ENA Express untuk lampiran antarmuka jaringan pada instans, instans pengirim memulai komunikasi dengan instans penerima, dan SRD mendeteksi apakah ENA Express beroperasi pada instans pengirim dan instans penerima. Jika ENA Ekspres beroperasi, komunikasi dapat menggunakan transmisi SRD. Jika ENA Ekspres tidak beroperasi, komunikasi kembali ke transmisi ENA standar.

Selama periode waktu ketika lalu lintas jaringan ringan, Anda mungkin melihat sedikit peningkatan latensi paket median (puluhan mikrodetik) ketika paket menggunakan ENA Express. Selama waktu tersebut, aplikasi yang memprioritaskan karakteristik performa jaringan tertentu dapat memperoleh manfaat dari ENA Ekspres sebagai berikut:

- Proses dapat memperoleh manfaat dari peningkatan bandwidth aliran tunggal maksimum dari 5 Gbps hingga 25 Gbps dalam Wilayah yang sama, hingga batas instans agregat. Misalnya, jika tipe instans tertentu mendukung hingga 12,5 Gbps, bandwidth aliran tunggal juga dibatasi hingga 12,5 Gbps.
- Proses yang berjalan lebih lama di Zona Ketersediaan yang sama akan mengalami pengurangan latensi ekor selama periode kemacetan jaringan.
- Proses dapat memperoleh manfaat dari distribusi yang lebih lancar dan lebih standar untuk waktu respons jaringan.

Topik


- Cara kerja ENA Ekspres
- Tipe instans yang didukung untuk ENA Ekspres
- Menyesuaikan kinerja untuk pengaturan ENA Express pada instans Linux
- Tinjau setelan ENA Express untuk instans EC2 Anda
- Mengonfigurasi pengaturan ENA Express untuk instans EC2 Anda


## Cara kerja ENA Ekspres

ENA Express didukung oleh teknologi AWS Scalable Reliable Datagram (SRD). Ini mendistribusikan paket untuk setiap aliran jaringan di jalur AWS jaringan yang berbeda, dan secara dinamis menyesuaikan distribusi ketika mendeteksi tanda-tanda kemacetan. Ini juga mengelola penataan ulang paket di ujung penerima.

Untuk memastikan bahwa ENA Ekspres dapat mengelola lalu lintas jaringan sebagaimana dimaksud, mengirim dan menerima instans dan komunikasi di antara mereka harus memenuhi semua persyaratan berikut:

- Baik tipe instans pengiriman maupun penerimaan didukung. Lihat tabel Tipe instans yang didukung untuk ENA Ekspres untuk informasi selengkapnya.
- Instans pengiriman dan penerimaan harus memiliki ENA Ekspres yang dikonfigurasi. Jika ada perbedaan dalam konfigurasi, Anda dapat mengalami situasi di mana lalu lintas default ke transmisi ENA standar. Skenario berikut menunjukkan apa yang bisa terjadi.

  Skenario: Perbedaan konfigurasi

| Instans | ENA Ekspres Diaktifkan | UDP menggunakan ENA Ekspres |
|---|---|---|
| Instans 1 | Ya | Ya |
| Instans 2 | Ya | Tidak |

  Dalam hal ini, lalu lintas TCP antara dua instans dapat menggunakan ENA Ekspres, karena kedua instans telah mengaktifkannya. Namun, karena salah satu instans tidak menggunakan

ENA Ekspres untuk lalu lintas UDP, komunikasi antara dua instans ini melalui UDP menggunakan transmisi ENA standar.

- Instans pengirim dan penerimaan harus berjalan di Wilayah yang sama.
- Jalur jaringan antara instans tidak boleh menyertakan kotak perangkat lunak perantara (middleware). ENA Ekspres saat ini tidak mendukung kotak perangkat lunak perantara (middleware).
- (Hanya instans Linux) Untuk memanfaatkan potensi bandwidth penuh, gunakan driver versi 2.2.9 atau lebih tinggi.
- (Hanya instans Linux) Untuk menghasilkan metrik, gunakan driver versi 2.8 atau lebih tinggi.

Jika ada persyaratan yang tidak terpenuhi, instans menggunakan TCP/UDP protokol standar tetapi tanpa SRD untuk berkomunikasi.

Untuk memastikan bahwa driver jaringan instans Anda dikonfigurasi untuk performa optimal, tinjau praktik terbaik yang disarankan untuk driver ENA. Praktik terbaik ini juga berlaku untuk ENA Ekspres. Untuk informasi selengkapnya, lihat Panduan Praktik Terbaik Driver Linux ENA dan Optimalisasi Kinerja di GitHub situs web.

> **Note**
>
> Amazon EC2 mengacu pada hubungan antara instans dan antarmuka jaringan yang melekat padanya sebagai lampiran. Pengaturan ENA Ekspres berlaku untuk lampiran. Jika antarmuka jaringan terlepas dari instans, lampiran tidak ada lagi, dan pengaturan ENA Ekspres yang diterapkan padanya tidak lagi berlaku. Hal yang sama berlaku ketika sebuah instans diakhiri, bahkan jika antarmuka jaringan tetap ada.

Setelah mengaktifkan ENA Express untuk lampiran antarmuka jaringan pada instans pengirim dan instans penerima, Anda dapat menggunakan metrik ENA Express untuk membantu memastikan bahwa instans Anda memanfaatkan sepenuhnya peningkatan kinerja yang disediakan teknologi SRD. Untuk informasi lebih lanjut tentang metrik ENA Ekspres, lihat Metrik untuk ENA Ekspres.

## Tipe instans yang didukung untuk ENA Ekspres

Jenis instance berikut mendukung ENA Express.

General purpose

| Tipe instans | Arsitektur |
| --- | --- |
| m6a.12xlarge | x86_64 |
| m6a.16xlarge | x86_64 |
| m6a.24xlarge | x86_64 |
| m6a.32xlarge | x86_64 |
| m6a.48xlarge | x86_64 |
| m6a.metal | x86_64 |
| m6i.8xlarge | x86_64 |
| m6i.12xlarge | x86_64 |
| m6i.16xlarge | x86_64 |
| m6i.24xlarge | x86_64 |
| m6i.32xlarge | x86_64 |
| m6i.metal | x86_64 |
| m6id.8xlarge | x86_64 |
| m6id.12xlarge | x86_64 |
| m6id.16xlarge | x86_64 |
| m6id.24xlarge | x86_64 |
| m6id.32xlarge | x86_64 |
| m6id.metal | x86_64 |
| m6idn.8xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| m6idn.12xlarge | x86_64 |
| m6idn.16xlarge | x86_64 |
| m6idn.24xlarge | x86_64 |
| m6idn.32xlarge | x86_64 |
| m6idn.metal | x86_64 |
| m6in.8xlarge | x86_64 |
| m6in.12xlarge | x86_64 |
| m6in.16xlarge | x86_64 |
| m6in.24xlarge | x86_64 |
| m6in.32xlarge | x86_64 |
| m6in.metal | x86_64 |
| m7a.12xlarge | x86_64 |
| m7a.16xlarge | x86_64 |
| m7a.24xlarge | x86_64 |
| m7a.32xlarge | x86_64 |
| m7a.48xlarge | x86_64 |
| m7a.metal-48xl | x86_64 |
| m7g.12xlarge | arm64 |
| m7g.16xlarge | arm64 |
| m7g.metal | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| m7gd.12xlarge | arm64 |
| m7gd.16xlarge | arm64 |
| m7gd.metal | arm64 |
| m7i.12xlarge | x86_64 |
| m7i.16xlarge | x86_64 |
| m7i.24xlarge | x86_64 |
| m7i.48xlarge | x86_64 |
| m7i.metal-24xl | x86_64 |
| m7i.metal-48xl | x86_64 |
| m8a.16xlarge | x86_64 |
| m8a.24xlarge | x86_64 |
| m8a.48xlarge | x86_64 |
| m8a.metal-24xl | x86_64 |
| m8a.metal-48xl | x86_64 |
| m8azn.12xlarge | x86_64 |
| m8azn.24xlarge | x86_64 |
| m8azn.metal-12xl | x86_64 |
| m8azn.metal-24xl | x86_64 |
| m8g.8xlarge | arm64 |
| m8g.12xlarge | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| m8g.16xlarge | arm64 |
| m8g.24xlarge | arm64 |
| m8g.48xlarge | arm64 |
| m8g.metal-24xl | arm64 |
| m8g.metal-48xl | arm64 |
| m8gb.4xlarge | arm64 |
| m8gb.8xlarge | arm64 |
| m8gb.12xlarge | arm64 |
| m8gb.16xlarge | arm64 |
| m8gb.24xlarge | arm64 |
| m8gb.48xlarge | arm64 |
| m8gb.metal-24xl | arm64 |
| m8gb.metal-48xl | arm64 |
| m8gd.8xlarge | arm64 |
| m8gd.12xlarge | arm64 |
| m8gd.16xlarge | arm64 |
| m8gd.24xlarge | arm64 |
| m8gd.48xlarge | arm64 |
| m8gd.metal-24xl | arm64 |
| m8gd.metal-48xl | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| m8gn.4xlarge | arm64 |
| m8gn.8xlarge | arm64 |
| m8gn.12xlarge | arm64 |
| m8gn.16xlarge | arm64 |
| m8gn.24xlarge | arm64 |
| m8gn.48xlarge | arm64 |
| m8gn.metal-24xl | arm64 |
| m8gn.metal-48xl | arm64 |
| m8i.8xlarge | x86_64 |
| m8i.12xlarge | x86_64 |
| m8i.16xlarge | x86_64 |
| m8i.24xlarge | x86_64 |
| m8i.32xlarge | x86_64 |
| m8i.48xlarge | x86_64 |
| m8i.96xlarge | x86_64 |
| m8i.metal-48xl | x86_64 |
| m8i.metal-96xl | x86_64 |
| m8id.8xlarge | x86_64 |
| m8id.12xlarge | x86_64 |
| m8id.16xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| m8id.24xlarge | x86_64 |
| m8id.32xlarge | x86_64 |
| m8id.48xlarge | x86_64 |
| m8id.96xlarge | x86_64 |
| m8id.metal-48xl | x86_64 |
| m8id.metal-96xl | x86_64 |
| m8in.8xlarge | x86_64 |
| m8in.12xlarge | x86_64 |
| m8in.16xlarge | x86_64 |
| m8in.24xlarge | x86_64 |
| m8in.32xlarge | x86_64 |
| m8in.48xlarge | x86_64 |
| m8in.96xlarge | x86_64 |
| m8in.metal-48xl | x86_64 |
| m8in.metal-96xl | x86_64 |
| m8idn.8xlarge | x86_64 |
| m8idn.12xlarge | x86_64 |
| m8idn.16xlarge | x86_64 |
| m8idn.24xlarge | x86_64 |
| m8idn.32xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| m8idn.48xlarge | x86_64 |
| m8idn.96xlarge | x86_64 |
| m8idn.metal-48xl | x86_64 |
| m8idn.metal-96xl | x86_64 |
| m8ib.8xlarge | x86_64 |
| m8ib.12xlarge | x86_64 |
| m8ib.16xlarge | x86_64 |
| m8ib.24xlarge | x86_64 |
| m8ib.32xlarge | x86_64 |
| m8ib.48xlarge | x86_64 |
| m8ib.96xlarge | x86_64 |
| m8ib.metal-48xl | x86_64 |
| m8ib.metal-96xl | x86_64 |
| m8idb.8xlarge | x86_64 |
| m8idb.12xlarge | x86_64 |
| m8idb.16xlarge | x86_64 |
| m8idb.24xlarge | x86_64 |
| m8idb.32xlarge | x86_64 |
| m8idb.48xlarge | x86_64 |
| m8idb.96xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| m8idb.metal-48xl | x86_64 |
| m8idb.metal-96xl | x86_64 |
| m9g.8xlarge | arm64 |
| m9g.12xlarge | arm64 |
| m9g.16xlarge | arm64 |
| m9g.24xlarge | arm64 |
| m9g.48xlarge | arm64 |
| m9g.metal-48xl | arm64 |
| m9gd.8xlarge | arm64 |
| m9gd.12xlarge | arm64 |
| m9gd.16xlarge | arm64 |
| m9gd.24xlarge | arm64 |
| m9gd.48xlarge | arm64 |
| m9gd.metal-48xl | arm64 |

Compute optimized

| Tipe instans | Arsitektur |
| --- | --- |
| c6a.12xlarge | x86_64 |
| c6a.16xlarge | x86_64 |
| c6a.24xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| c6a.32xlarge | x86_64 |
| c6a.48xlarge | x86_64 |
| c6a.metal | x86_64 |
| c6gn.4xlarge | arm64 |
| c6gn.8xlarge | arm64 |
| c6gn.12xlarge | arm64 |
| c6gn.16xlarge | arm64 |
| c6i.8xlarge | x86_64 |
| c6i.12xlarge | x86_64 |
| c6i.16xlarge | x86_64 |
| c6i.24xlarge | x86_64 |
| c6i.32xlarge | x86_64 |
| c6i.metal | x86_64 |
| c6id.8xlarge | x86_64 |
| c6id.12xlarge | x86_64 |
| c6id.16xlarge | x86_64 |
| c6id.24xlarge | x86_64 |
| c6id.32xlarge | x86_64 |
| c6id.metal | x86_64 |
| c6in.8xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| c6in.12xlarge | x86_64 |
| c6in.16xlarge | x86_64 |
| c6in.24xlarge | x86_64 |
| c6in.32xlarge | x86_64 |
| c6in.metal | x86_64 |
| c7a.12xlarge | x86_64 |
| c7a.16xlarge | x86_64 |
| c7a.24xlarge | x86_64 |
| c7a.32xlarge | x86_64 |
| c7a.48xlarge | x86_64 |
| c7a.metal-48xl | x86_64 |
| c7g.12xlarge | arm64 |
| c7g.16xlarge | arm64 |
| c7g.metal | arm64 |
| c7gd.12xlarge | arm64 |
| c7gd.16xlarge | arm64 |
| c7gd.metal | arm64 |
| c7gn.4xlarge | arm64 |
| c7gn.8xlarge | arm64 |
| c7gn.12xlarge | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| c7gn.16xlarge | arm64 |
| c7gn.metal | arm64 |
| c7i.12xlarge | x86_64 |
| c7i.16xlarge | x86_64 |
| c7i.24xlarge | x86_64 |
| c7i.48xlarge | x86_64 |
| c7i.metal-24xl | x86_64 |
| c7i.metal-48xl | x86_64 |
| c8a.16xlarge | x86_64 |
| c8a.24xlarge | x86_64 |
| c8a.48xlarge | x86_64 |
| c8a.metal-24xl | x86_64 |
| c8a.metal-48xl | x86_64 |
| c8g.8xlarge | arm64 |
| c8g.12xlarge | arm64 |
| c8g.16xlarge | arm64 |
| c8g.24xlarge | arm64 |
| c8g.48xlarge | arm64 |
| c8g.metal-24xl | arm64 |
| c8g.metal-48xl | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| c8gb.4xlarge | arm64 |
| c8gb.8xlarge | arm64 |
| c8gb.12xlarge | arm64 |
| c8gb.16xlarge | arm64 |
| c8gb.24xlarge | arm64 |
| c8gb.48xlarge | arm64 |
| c8gb.metal-24xl | arm64 |
| c8gb.metal-48xl | arm64 |
| c8gd.8xlarge | arm64 |
| c8gd.12xlarge | arm64 |
| c8gd.16xlarge | arm64 |
| c8gd.24xlarge | arm64 |
| c8gd.48xlarge | arm64 |
| c8gd.metal-24xl | arm64 |
| c8gd.metal-48xl | arm64 |
| c8gn.4xlarge | arm64 |
| c8gn.8xlarge | arm64 |
| c8gn.12xlarge | arm64 |
| c8gn.16xlarge | arm64 |
| c8gn.24xlarge | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| c8gn.48xlarge | arm64 |
| c8gn.metal-24xl | arm64 |
| c8gn.metal-48xl | arm64 |
| c8i.8xlarge | x86_64 |
| c8i.12xlarge | x86_64 |
| c8i.16xlarge | x86_64 |
| c8i.24xlarge | x86_64 |
| c8i.32xlarge | x86_64 |
| c8i.48xlarge | x86_64 |
| c8i.96xlarge | x86_64 |
| c8i.metal-48xl | x86_64 |
| c8i.metal-96xl | x86_64 |
| c8id.8xlarge | x86_64 |
| c8id.12xlarge | x86_64 |
| c8id.16xlarge | x86_64 |
| c8id.24xlarge | x86_64 |
| c8id.32xlarge | x86_64 |
| c8id.48xlarge | x86_64 |
| c8id.96xlarge | x86_64 |
| c8id.metal-48xl | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| c8id.metal-96xl | x86_64 |
| c8in.8xlarge | x86_64 |
| c8in.12xlarge | x86_64 |
| c8in.16xlarge | x86_64 |
| c8in.24xlarge | x86_64 |
| c8in.32xlarge | x86_64 |
| c8in.48xlarge | x86_64 |
| c8in.96xlarge | x86_64 |
| c8in.metal-48xl | x86_64 |
| c8in.metal-96xl | x86_64 |
| c8ib.8xlarge | x86_64 |
| c8ib.12xlarge | x86_64 |
| c8ib.16xlarge | x86_64 |
| c8ib.24xlarge | x86_64 |
| c8ib.32xlarge | x86_64 |
| c8ib.48xlarge | x86_64 |
| c8ib.96xlarge | x86_64 |
| c8ib.metal-48xl | x86_64 |
| c8ib.metal-96xl | x86_64 |
| c9g.8xlarge | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| c9g.12xlarge | arm64 |
| c9g.16xlarge | arm64 |
| c9g.24xlarge | arm64 |
| c9g.48xlarge | arm64 |
| c9g.metal-48xl | arm64 |
| c9gd.8xlarge | arm64 |
| c9gd.12xlarge | arm64 |
| c9gd.16xlarge | arm64 |
| c9gd.24xlarge | arm64 |
| c9gd.48xlarge | arm64 |
| c9gd.metal-48xl | arm64 |

Memory optimized

| Tipe instans | Arsitektur |
| --- | --- |
| r6a.12xlarge | x86_64 |
| r6a.16xlarge | x86_64 |
| r6a.24xlarge | x86_64 |
| r6a.32xlarge | x86_64 |
| r6a.48xlarge | x86_64 |
| r6a.metal | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r6i.8xlarge | x86_64 |
| r6i.12xlarge | x86_64 |
| r6i.16xlarge | x86_64 |
| r6i.24xlarge | x86_64 |
| r6i.32xlarge | x86_64 |
| r6i.metal | x86_64 |
| r6id.8xlarge | x86_64 |
| r6id.12xlarge | x86_64 |
| r6id.16xlarge | x86_64 |
| r6id.24xlarge | x86_64 |
| r6id.32xlarge | x86_64 |
| r6id.metal | x86_64 |
| r6idn.8xlarge | x86_64 |
| r6idn.12xlarge | x86_64 |
| r6idn.16xlarge | x86_64 |
| r6idn.24xlarge | x86_64 |
| r6idn.32xlarge | x86_64 |
| r6idn.metal | x86_64 |
| r6in.8xlarge | x86_64 |
| r6in.12xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r6in.16xlarge | x86_64 |
| r6in.24xlarge | x86_64 |
| r6in.32xlarge | x86_64 |
| r6in.metal | x86_64 |
| r7a.12xlarge | x86_64 |
| r7a.16xlarge | x86_64 |
| r7a.24xlarge | x86_64 |
| r7a.32xlarge | x86_64 |
| r7a.48xlarge | x86_64 |
| r7a.metal-48xl | x86_64 |
| r7g.12xlarge | arm64 |
| r7g.16xlarge | arm64 |
| r7g.metal | arm64 |
| r7gd.12xlarge | arm64 |
| r7gd.16xlarge | arm64 |
| r7gd.metal | arm64 |
| r7i.12xlarge | x86_64 |
| r7i.16xlarge | x86_64 |
| r7i.24xlarge | x86_64 |
| r7i.48xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r7i.metal-24xl | x86_64 |
| r7i.metal-48xl | x86_64 |
| r7iz.8xlarge | x86_64 |
| r7iz.12xlarge | x86_64 |
| r7iz.16xlarge | x86_64 |
| r7iz.32xlarge | x86_64 |
| r7iz.metal-16xl | x86_64 |
| r7iz.metal-32xl | x86_64 |
| r8a.16xlarge | x86_64 |
| r8a.24xlarge | x86_64 |
| r8a.48xlarge | x86_64 |
| r8a.metal-24xl | x86_64 |
| r8a.metal-48xl | x86_64 |
| r8g.8xlarge | arm64 |
| r8g.12xlarge | arm64 |
| r8g.16xlarge | arm64 |
| r8g.24xlarge | arm64 |
| r8g.48xlarge | arm64 |
| r8g.metal-24xl | arm64 |
| r8g.metal-48xl | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r8gb.4xlarge | arm64 |
| r8gb.8xlarge | arm64 |
| r8gb.12xlarge | arm64 |
| r8gb.16xlarge | arm64 |
| r8gb.24xlarge | arm64 |
| r8gb.48xlarge | arm64 |
| r8gb.metal-24xl | arm64 |
| r8gb.metal-48xl | arm64 |
| r8gd.8xlarge | arm64 |
| r8gd.12xlarge | arm64 |
| r8gd.16xlarge | arm64 |
| r8gd.24xlarge | arm64 |
| r8gd.48xlarge | arm64 |
| r8gd.metal-24xl | arm64 |
| r8gd.metal-48xl | arm64 |
| r8gn.4xlarge | arm64 |
| r8gn.8xlarge | arm64 |
| r8gn.12xlarge | arm64 |
| r8gn.16xlarge | arm64 |
| r8gn.24xlarge | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r8gn.48xlarge | arm64 |
| r8gn.metal-24xl | arm64 |
| r8gn.metal-48xl | arm64 |
| r8i.8xlarge | x86_64 |
| r8i.12xlarge | x86_64 |
| r8i.16xlarge | x86_64 |
| r8i.24xlarge | x86_64 |
| r8i.32xlarge | x86_64 |
| r8i.48xlarge | x86_64 |
| r8i.96xlarge | x86_64 |
| r8i.metal-48xl | x86_64 |
| r8i.metal-96xl | x86_64 |
| r8id.8xlarge | x86_64 |
| r8id.12xlarge | x86_64 |
| r8id.16xlarge | x86_64 |
| r8id.24xlarge | x86_64 |
| r8id.32xlarge | x86_64 |
| r8id.48xlarge | x86_64 |
| r8id.96xlarge | x86_64 |
| r8id.metal-48xl | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r8id.metal-96xl | x86_64 |
| r8in.8xlarge | x86_64 |
| r8in.12xlarge | x86_64 |
| r8in.16xlarge | x86_64 |
| r8in.24xlarge | x86_64 |
| r8in.32xlarge | x86_64 |
| r8in.48xlarge | x86_64 |
| r8in.96xlarge | x86_64 |
| r8in.metal-48xl | x86_64 |
| r8in.metal-96xl | x86_64 |
| r8idn.8xlarge | x86_64 |
| r8idn.12xlarge | x86_64 |
| r8idn.16xlarge | x86_64 |
| r8idn.24xlarge | x86_64 |
| r8idn.32xlarge | x86_64 |
| r8idn.48xlarge | x86_64 |
| r8idn.96xlarge | x86_64 |
| r8idn.metal-48xl | x86_64 |
| r8idn.metal-96xl | x86_64 |
| r8ib.8xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r8ib.12xlarge | x86_64 |
| r8ib.16xlarge | x86_64 |
| r8ib.24xlarge | x86_64 |
| r8ib.32xlarge | x86_64 |
| r8ib.48xlarge | x86_64 |
| r8ib.96xlarge | x86_64 |
| r8ib.metal-48xl | x86_64 |
| r8ib.metal-96xl | x86_64 |
| r8idb.8xlarge | x86_64 |
| r8idb.12xlarge | x86_64 |
| r8idb.16xlarge | x86_64 |
| r8idb.24xlarge | x86_64 |
| r8idb.32xlarge | x86_64 |
| r8idb.48xlarge | x86_64 |
| r8idb.96xlarge | x86_64 |
| r8idb.metal-48xl | x86_64 |
| r8idb.metal-96xl | x86_64 |
| r9g.8xlarge | arm64 |
| r9g.12xlarge | arm64 |
| r9g.16xlarge | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| r9g.24xlarge | arm64 |
| r9g.48xlarge | arm64 |
| r9g.metal-48xl | arm64 |
| r9gd.8xlarge | arm64 |
| r9gd.12xlarge | arm64 |
| r9gd.16xlarge | arm64 |
| r9gd.24xlarge | arm64 |
| r9gd.48xlarge | arm64 |
| r9gd.metal-48xl | arm64 |
| u7i-6tb.112xlarge | x86_64 |
| u7i-8tb.112xlarge | x86_64 |
| u7i-12tb.224xlarge | x86_64 |
| u7in-16tb.224xlarge | x86_64 |
| u7in-24tb.224xlarge | x86_64 |
| u7in-32tb.224xlarge | x86_64 |
| u7inh-32tb.480xlarge | x86_64 |
| x2idn.16xlarge | x86_64 |
| x2idn.24xlarge | x86_64 |
| x2idn.32xlarge | x86_64 |
| x2idn.metal | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| x2iedn.8xlarge | x86_64 |
| x2iedn.16xlarge | x86_64 |
| x2iedn.24xlarge | x86_64 |
| x2iedn.32xlarge | x86_64 |
| x2iedn.metal | x86_64 |
| x8g.8xlarge | arm64 |
| x8g.12xlarge | arm64 |
| x8g.16xlarge | arm64 |
| x8g.24xlarge | arm64 |
| x8g.48xlarge | arm64 |
| x8g.metal-24xl | arm64 |
| x8g.metal-48xl | arm64 |
| x8aedz.24xlarge | x86_64 |
| x8aedz.metal-24xl | x86_64 |
| x8i.8xlarge | x86_64 |
| x8i.12xlarge | x86_64 |
| x8i.16xlarge | x86_64 |
| x8i.24xlarge | x86_64 |
| x8i.32xlarge | x86_64 |
| x8i.48xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| x8i.64xlarge | x86_64 |
| x8i.96xlarge | x86_64 |
| x8i.metal-48xl | x86_64 |
| x8i.metal-96xl | x86_64 |

Accelerated computing

| Tipe instans | Arsitektur |
| --- | --- |
| g6.48xlarge | x86_64 |
| g6e.12xlarge | x86_64 |
| g6e.24xlarge | x86_64 |
| g6e.48xlarge | x86_64 |
| g7.24xlarge | x86_64 |
| g7.48xlarge | x86_64 |
| g7e.12xlarge | x86_64 |
| g7e.24xlarge | x86_64 |
| g7e.48xlarge | x86_64 |
| p5.4xlarge | x86_64 |
| p5.48xlarge | x86_64 |
| p5e.48xlarge | x86_64 |
| p5en.48xlarge | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| p6-b200.48xlarge | x86_64 |
| p6-b300.48xlarge | x86_64 |
| p6e-gb200.36xlarge | arm64 |

Storage optimized

| Tipe instans | Arsitektur |
| --- | --- |
| i4g.4xlarge | arm64 |
| i4g.8xlarge | arm64 |
| i4g.16xlarge | arm64 |
| i4i.8xlarge | x86_64 |
| i4i.12xlarge | x86_64 |
| i4i.16xlarge | x86_64 |
| i4i.24xlarge | x86_64 |
| i4i.32xlarge | x86_64 |
| i4i.metal | x86_64 |
| i7i.12xlarge | x86_64 |
| i7i.16xlarge | x86_64 |
| i7i.24xlarge | x86_64 |
| i7i.48xlarge | x86_64 |
| i7i.metal-24xl | x86_64 |

| Tipe instans | Arsitektur |
| --- | --- |
| i7i.metal-48xl | x86_64 |
| i7ie.6xlarge | x86_64 |
| i7ie.12xlarge | x86_64 |
| i7ie.18xlarge | x86_64 |
| i7ie.24xlarge | x86_64 |
| i7ie.48xlarge | x86_64 |
| i7ie.metal-24xl | x86_64 |
| i7ie.metal-48xl | x86_64 |
| i8g.8xlarge | arm64 |
| i8g.12xlarge | arm64 |
| i8g.16xlarge | arm64 |
| i8g.24xlarge | arm64 |
| i8g.48xlarge | arm64 |
| i8g.metal-24xl | arm64 |
| i8g.metal-48xl | arm64 |
| i8ge.6xlarge | arm64 |
| i8ge.12xlarge | arm64 |
| i8ge.18xlarge | arm64 |
| i8ge.24xlarge | arm64 |
| i8ge.48xlarge | arm64 |

| Tipe instans | Arsitektur |
| --- | --- |
| i8ge.metal-24xl | arm64 |
| i8ge.metal-48xl | arm64 |
| im4gn.4xlarge | arm64 |
| im4gn.8xlarge | arm64 |
| im4gn.16xlarge | arm64 |

## Menyesuaikan kinerja untuk pengaturan ENA Express pada instans Linux

Untuk memastikan bahwa ENA Express dapat beroperasi secara efektif, instance Linux Anda harus memenuhi beberapa persyaratan konfigurasi jaringan.

Daripada mengonfigurasi setiap pengaturan secara manual, Anda dapat mengunduh dan menjalankan skrip pemeriksaan pengaturan ENA Express dari GitHub repositori Amazon. Skrip memvalidasi instance Anda terhadap pengaturan yang diperlukan dan direkomendasikan untuk ENA Express, dan menampilkan perintah yang tepat untuk memperbaiki masalah yang ditemukannya.

https://github.com/amzn/amzn-ec2-ena-utilities/blob/main/ena-express/check-ena-express-settings.sh

Skrip memeriksa pengaturan dan konfigurasi berikut:

- Ukuran MTU - ENA Express memerlukan MTU yang lebih rendah daripada default untuk mengakomodasi header AWS SRD tambahan. Koneksi TCP yang baru dibuat secara otomatis menjepit MSS untuk mengurangi ini, tetapi lalu lintas UDP masih membutuhkan MTU yang lebih rendah.
- Batas ukuran antrian keluaran TCP — Memeriksa apakah batas byte dalam penerbangan per soket cukup untuk mempertahankan throughput yang tinggi. Lingkungan dengan latensi jaringan yang meningkat memerlukan batas yang lebih tinggi.
- Batas antrian byte — Mengkonfirmasi bahwa batas antrian byte (BQL) dinonaktifkan pada antarmuka jaringan. BQL dapat membatasi jumlah data yang diantri untuk transmisi tingkat perangkat, yang membatasi kinerja ENA Express.

> **Note**
>
> Driver ENA untuk distribusi Amazon Linux menonaktifkan batas antrean byte secara default.

- TCP autocorking — Memeriksa apakah autocorking TCP dinonaktifkan. Menonaktifkan autocorking dapat mengurangi latensi untuk pola lalu lintas TCP ENA Express tertentu, seperti beban kerja respons permintaan. Hal ini dapat mengakibatkan peningkatan minimal dalam overhead pemrosesan paket.
- Ukuran antrian TX dan LLQ Besar — Memverifikasi bahwa ukuran antrian transmisi untuk antarmuka jaringan cukup besar untuk kinerja optimal. Skrip juga memeriksa apakah parameter modul ENA secara eksplisit menonaktifkan fitur Large Low Latency Queue (Large Low Latency Queue), karena dapat mengurangi kedalaman antrian TX yang tersedia. Untuk informasi selengkapnya tentang Large LLQ dan dampaknya terhadap ukuran antrian TX, lihat Antrian Latensi Low Large (Large LLQ) aktif. GitHub
- Ukuran antrian RX — Memeriksa apakah buffer cincin penerima untuk antarmuka jaringan cukup besar untuk menangani lalu lintas masuk secara efisien dan menghindari penurunan paket di bawah beban.
- Ukuran buffer soket TCP dan jaringan — Memvalidasi bahwa ukuran maksimum buffer penerima dan kirim TCP, serta default dan maksimum buffer soket jaringan inti, cukup besar untuk mempertahankan throughput tinggi. Pengaturan ini penting di lingkungan dengan latensi jaringan yang meningkat, di mana Anda memerlukan buffer yang lebih besar untuk menggunakan koneksi.
- Kontrol kemacetan TCP — Memverifikasi bahwa konfigurasi kontrol kemacetan TCP dioptimalkan untuk digunakan dengan ENA Express di lingkungan dengan latensi jaringan yang meningkat.

Skrip juga melaporkan informasi diagnostik tambahan, termasuk versi driver ENA, statistik SRD ENA, pengaturan moderasi interupsi, konfigurasi antrian, dan ukuran buffer soket. Informasi ini dapat berguna untuk memecahkan masalah kinerja ENA Express.

Untuk memastikan bahwa driver jaringan instans Anda dikonfigurasi untuk kinerja optimal, tinjau juga Panduan Optimasi Kinerja dan Praktik Terbaik Driver ENA Linux diGitHub.

## Tinjau setelan ENA Express untuk instans EC2 Anda

Anda dapat memverifikasi pengaturan ENA Express berdasarkan instance atau dengan antarmuka jaringan. Untuk memperbarui pengaturan ENA Express, lihatthe section called “Konfigurasikan pengaturan instance”.

### Console

Untuk melihat pengaturan ENA Express untuk antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Antarmuka jaringan.
3. Pilih antarmuka jaringan untuk melihat detail untuk instans itu. Anda dapat memilih tautan ID antarmuka Jaringan untuk membuka halaman detail, atau Anda dapat memilih kotak centang di sisi kiri daftar.
4. Di bagian Lampiran antarmuka Jaringan pada tab Detail atau halaman detail, tinjau pengaturan untuk ENA Express dan ENA Express UDP.

Untuk melihat setelan ENA Express untuk sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans untuk melihat detail untuk instans itu. Anda dapat memilih tautan ID Instans untuk membuka halaman detail, atau Anda dapat memilih kotak centang di sisi kiri daftar untuk melihat detail di panel detail di bagian bawah halaman.
4. Di bagian Antarmuka jaringan pada tab Jaringan, gulir ke kanan untuk meninjau pengaturan untuk ENA Ekspres dan UDP ENA Ekspres.

### AWS CLI

Untuk mendapatkan pengaturan ENA Express untuk sebuah instance

Gunakan perintah describe-instances. Contoh perintah ini mengembalikan daftar konfigurasi ENA Express untuk antarmuka jaringan yang dilampirkan ke masing-masing instance berjalan yang ditentukan oleh `--instance-ids` parameter.

```
aws ec2 describe-instances \
```

```
    --instance-ids i-1234567890abcdef0 i-0598c7d356eba48d7 \
    --query 'Reservations[*].Instances[*].[InstanceId,
 NetworkInterfaces[*].Attachment.EnaSrdSpecification]'
```

Berikut ini adalah output contoh.

```
[
    [
        [
            "i-1234567890abcdef0",
            [
                {
                    "EnaSrdEnabled": true,
                    "EnaSrdUdpSpecification": {
                        "EnaSrdUdpEnabled": false
                    }
                }
            ]
        ]
    ],
    [
        [
            "i-0598c7d356eba48d7",
            [
            {
                    "EnaSrdEnabled": true,
                    "EnaSrdUdpSpecification": {
                        "EnaSrdUdpEnabled": false
                    }
                }
            ]
        ]
    ]
]
```

Untuk mendapatkan pengaturan ENA Express untuk antarmuka jaringan

Gunakan perintah describe-network-interfaces.

```
aws ec2 describe-network-interfaces \
    --network-interface-ids eni-1234567890abcdef0 \
    --query NetworkInterfaces[].[NetworkInterfaceId,Attachment.EnaSrdSpecification]
```

Berikut ini adalah output contoh.

```
[
    [
        "eni-1234567890abcdef0",
        {
            "EnaSrdEnabled": true,
            "EnaSrdUdpSpecification": {
                "EnaSrdUdpEnabled": false
            }
        }
    ]
]
```

PowerShell

Untuk mendapatkan pengaturan ENA Express untuk antarmuka jaringan

Gunakan Get-EC2NetworkInterface cmdlet.

```
Get-EC2NetworkInterface `
    -NetworkInterfaceId eni-1234567890abcdef0 | `
Select-Object `
    Association,
    NetworkInterfaceId,
    OwnerId,
    @{Name = 'AttachTime'; Expression = { $_.Attachment.AttachTime } },
    @{Name = 'AttachmentId'; Expression = { $_.Attachment.AttachmentId } },
    @{Name = 'DeleteOnTermination'; Expression =
 { $_.Attachment.DeleteOnTermination } },
    @{Name = 'NetworkCardIndex'; Expression = { $_.Attachment.NetworkCardIndex } },
    @{Name = 'InstanceId'; Expression = { $_.Attachment.InstanceId } },
    @{Name = 'InstanceOwnerId'; Expression = { $_.Attachment.InstanceOwnerId } },
    @{Name = 'Status'; Expression = { $_.Attachment.Status } },
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled } },
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled } }
```

Berikut ini adalah output contoh.

```
Association         :
```

```
NetworkInterfaceId  : eni-0d1234e5f6a78901b
OwnerId             : 111122223333
AttachTime          : 6/11/2022 1:13:11 AM
AttachmentId        : eni-attach-0d1234e5f6a78901b
DeleteOnTermination : True
NetworkCardIndex    : 0
InstanceId          : i-1234567890abcdef0
InstanceOwnerId     : 111122223333
Status              : attached
EnaSrdEnabled       : True
EnaSrdUdpEnabled    : False
```

## Mengonfigurasi pengaturan ENA Express untuk instans EC2 Anda

Anda dapat mengonfigurasi ENA Ekspres untuk tipe instans EC2 yang didukung tanpa perlu menginstal perangkat lunak tambahan apa pun. Untuk informasi selengkapnya, lihat the section called “Tipe instans yang didukung untuk ENA Ekspres”.

Console

Untuk mengelola ENA Express untuk antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Antarmuka jaringan.
3. Pilih antarmuka jaringan yang dilampirkan ke sebuah instans. Anda dapat memilih tautan ID antarmuka Jaringan untuk membuka halaman detail, atau Anda dapat memilih kotak centang di sisi kiri daftar.
4. Pilih Kelola ENA Ekspres dari menu Tindakan di sisi kanan atas halaman. Ini membuka dialog Kelola ENA Ekspres, dengan ID antarmuka jaringan yang dipilih dan pengaturan saat ini ditampilkan.

   Jika antarmuka jaringan yang Anda pilih tidak dilampirkan ke sebuah instans, tindakan ini tidak muncul di menu.
5. Untuk menggunakan ENA Express, pilih kotak centang Aktifkan.
6. Ketika ENA Ekspres diaktifkan, Anda dapat mengonfigurasi pengaturan UDP. Untuk menggunakan ENA Express UDP, pilih kotak centang Aktifkan.
7. Untuk menyimpan pengaturan Anda, pilih Simpan.

Untuk mengelola ENA Express untuk sebuah contoh

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans yang ingin Anda kelola. Anda dapat memilih ID Instans untuk membuka halaman detail, atau Anda dapat memilih kotak centang di sisi kiri daftar.
4. Pilih antarmuka Jaringan yang akan dikonfigurasi untuk instans Anda.
5. Pilih Kelola ENA Ekspres dari menu Tindakan di sisi kanan atas halaman.
6. Untuk mengonfigurasi ENA Ekspres untuk antarmuka jaringan yang dilampirkan ke instans Anda, pilih dari daftar antarmuka Jaringan.
7. Untuk menggunakan ENA Express untuk lampiran antarmuka jaringan yang dipilih, pilih kotak centang Aktifkan.
8. Ketika ENA Ekspres diaktifkan, Anda dapat mengonfigurasi pengaturan UDP. Untuk menggunakan ENA Express UDP, pilih kotak centang Aktifkan.
9. Untuk menyimpan pengaturan Anda, pilih Simpan.

Untuk mengkonfigurasi ENA Express saat Anda melampirkan antarmuka jaringan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Antarmuka jaringan.
3. Pilih antarmuka jaringan yang tidak dilampirkan ke instans (Status Tersedia). Anda dapat memilih tautan ID antarmuka Jaringan untuk membuka halaman detail, atau Anda dapat memilih kotak centang di sisi kiri daftar.
4. Pilih Instans yang akan Anda lampirkan.
5. Untuk menggunakan ENA Express setelah Anda melampirkan antarmuka jaringan ke instance, pilih kotak centang Aktifkan.
6. Ketika ENA Ekspres diaktifkan, Anda dapat mengonfigurasi pengaturan UDP. Untuk menggunakan ENA Express UDP, pilih kotak centang Aktifkan.
7. Untuk melampirkan antarmuka jaringan ke instans dan menyimpan pengaturan ENA Ekspres Anda, pilih Lampirkan.

AWS CLI

Untuk mengkonfigurasi ENA Express saat Anda melampirkan antarmuka jaringan

Gunakan attach-network-interface perintah, seperti yang ditunjukkan pada contoh berikut.

Contoh 1: Gunakan ENA Express untuk lalu lintas TCP, tetapi bukan lalu lintas UDP

Contoh ini mengonfigurasi `EnaSrdEnabled` sebagai`true`, dan memungkinkan `EnaSrdUdpEnabled` untuk default ke`false`.

```
aws ec2 attach-network-interface \
    --network-interface-id eni-1234567890abcdef0 \
    --instance-id i-1234567890abcdef0 \
    --device-index 1 \
    --ena-srd-specification 'EnaSrdEnabled=true'
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Contoh ini mengonfigurasi keduanya `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai`true`.

```
aws ec2 attach-network-interface \
    --network-interface-id eni-1234567890abcdef0 \
    --instance-id i-1234567890abcdef0 \
    --device-index 1 \
    --ena-srd-specification
 'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'
```

Untuk memperbarui pengaturan ENA Express untuk lampiran antarmuka jaringan

Gunakan modify-network-interface-attribute perintah seperti yang ditunjukkan dalam contoh berikut.

Contoh 1: Gunakan ENA Express untuk lalu lintas TCP, tetapi bukan lalu lintas UDP

Contoh ini mengonfigurasi `EnaSrdEnabled` sebagai`true`, dan memungkinkan `EnaSrdUdpEnabled` untuk default ke `false` jika belum pernah disetel sebelumnya.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
    --ena-srd-specification 'EnaSrdEnabled=true'
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Contoh ini mengonfigurasi keduanya `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai`true`.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
    --ena-srd-specification
 'EnaSrdEnabled=true,EnaSrdUdpSpecification={EnaSrdUdpEnabled=true}'
```

Contoh 3: Berhenti menggunakan ENA Ekspres untuk lalu lintas UDP

Contoh ini mengkonfigurasi `EnaSrdUdpEnabled` sebagai`false`.

```
aws ec2 modify-network-interface-attribute \
    --network-interface-id eni-1234567890abcdef0 \
    --ena-srd-specification 'EnaSrdUdpSpecification={EnaSrdUdpEnabled=false}'
```

PowerShell

Untuk mengkonfigurasi ENA Express saat Anda melampirkan antarmuka jaringan

Gunakan Add-EC2NetworkInterface cmdlet seperti yang ditunjukkan dalam contoh berikut.

Contoh 1: Gunakan ENA Express untuk lalu lintas TCP, tetapi bukan lalu lintas UDP

Contoh ini mengonfigurasi `EnaSrdEnabled` sebagai`true`, dan memungkinkan `EnaSrdUdpEnabled` untuk default ke`false`.

```
Add-EC2NetworkInterface `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -InstanceId i-1234567890abcdef0 `
    -DeviceIndex 1 `
    -EnaSrdSpecification_EnaSrdEnabled $true
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Contoh ini mengonfigurasi keduanya `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai`true`.

```
Add-EC2NetworkInterface `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -InstanceId i-1234567890abcdef0 `
    -DeviceIndex 1 `
    -EnaSrdSpecification_EnaSrdEnabled $true `
    -EnaSrdUdpSpecification_EnaSrdUdpEnabled $true
```

Untuk mengkonfigurasi pengaturan ENA Express untuk lampiran antarmuka jaringan Anda

Gunakan Edit-EC2NetworkInterfaceAttribute cmdlet seperti yang ditunjukkan dalam contoh berikut.

Contoh 1: Gunakan ENA Express untuk lalu lintas TCP, tetapi bukan lalu lintas UDP

Contoh ini mengonfigurasi `EnaSrdEnabled` sebagai`true`, dan memungkinkan `EnaSrdUdpEnabled` untuk default ke `false` jika belum pernah disetel sebelumnya.

```
Edit-EC2NetworkInterfaceAttribute `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -EnaSrdSpecification_EnaSrdEnabled $true ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
    NetworkInterfaceId,
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List
```

Contoh 2: Gunakan ENA Ekspres untuk lalu lintas TCP dan lalu lintas UDP

Contoh ini mengonfigurasi keduanya `EnaSrdEnabled` dan `EnaSrdUdpEnabled` sebagai`true`.

```
Edit-EC2NetworkInterfaceAttribute `
    -NetworkInterfaceId eni-1234567890abcdef0 `
    -EnaSrdSpecification_EnaSrdEnabled $true `
    -EnaSrdSpecification_EnaSrdUdpSpecification_EnaSrdUdpEnabled $true ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-1234567890abcdef0 | `
Select-Object `
    NetworkInterfaceId,
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List
```

Contoh 3: Berhenti menggunakan ENA Ekspres untuk lalu lintas UDP

Contoh ini mengkonfigurasi `EnaSrdUdpEnabled` sebagai`false`.

```
Edit-EC2NetworkInterfaceAttribute `
```

```
    -NetworkInterfaceId eni-0123f4567890a1b23 `
    -EnaSrdSpecification_EnaSrdUdpSpecification_EnaSrdUdpEnabled $false ;
Get-EC2NetworkInterface -NetworkInterfaceId eni-0123f4567890a1b23 | `
Select-Object `
    NetworkInterfaceId,
    @{Name = 'EnaSrdEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdEnabled }},
    @{Name = 'EnaSrdUdpEnabled'; Expression =
 { $_.Attachment.EnaSrdSpecification.EnaSrdUdpSpecification.EnaSrdUdpEnabled }} | `
Format-List
```

Konfigurasikan ENA Express saat peluncuran

Anda dapat menggunakan salah satu metode berikut untuk mengkonfigurasi ENA Express secara langsung saat Anda meluncurkan instance. Tautan yang ditentukan merujuk Anda ke Konsol Manajemen AWS instruksi untuk metode ini.

- Panduan peluncuran instans — Anda dapat mengonfigurasi ENA Express saat diluncurkan dengan wizard instance peluncuran. Untuk informasi selengkapnya, lihat Konfigurasi jaringan lanjutan di wizard Pengaturan jaringan untuk instance peluncuran.
- Template peluncuran - Anda dapat mengonfigurasi ENA Express saat peluncuran saat Anda menggunakan template peluncuran. Untuk informasi selengkapnya, lihat Buat template peluncuran Amazon EC2 halaman, lalu perluas bagian Pengaturan jaringan dan tinjau konfigurasi jaringan lanjutan.

## Jaringan yang ditingkatkan dengan antarmuka Intel 82599 VF

Misalnya Xen-based , antarmuka Intel 82599 Virtual Function (VF) menyediakan kemampuan jaringan yang ditingkatkan. Antarmuka menggunakan `ixgbevf` driver Intel.

Tab berikut menunjukkan cara memverifikasi driver adaptor jaringan yang diinstal untuk sistem operasi instans Anda.

Linux

Driver antarmuka jaringan Linux

Gunakan perintah berikut untuk memverifikasi apakah modul sedang digunakan pada antarmuka tertentu, menggantikan nama antarmuka yang ingin Anda periksa. Jika Anda menggunakan

antarmuka tunggal (default), ini adalah `eth0`. Jika sistem operasi mendukung nama jaringan yang dapat diprediksi, ini bisa menjadi nama seperti `ens5`.

Dalam contoh berikut, modul `ixgbevf` tidak dimuat, karena driver yang terdaftar adalah `vif`.

```
[ec2-user ~]$ ethtool -i eth0
driver: vif
version:
firmware-version:
bus-info: vif-0
supports-statistics: yes
supports-test: no
supports-eeprom-access: no
supports-register-dump: no
supports-priv-flags: no
```

Dalam contoh ini, modul `ixgbevf` dimuat. Instans ini memiliki jaringan yang ditingkatkan, yang dikonfigurasi dengan benar.

```
[ec2-user ~]$ ethtool -i eth0
driver: ixgbevf
version: 4.0.3
firmware-version: N/A
bus-info: 0000:00:03.0
supports-statistics: yes
supports-test: yes
supports-eeprom-access: no
supports-register-dump: yes
supports-priv-flags: no
```

Windows

Adaptor jaringan Windows

Untuk memverifikasi bahwa driver telah diinstal, sambungkan ke instans Anda dan buka Pengelola Perangkat. Anda akan melihat `Intel(R) 82599 Virtual Function` tercantum di bawah Adaptor jaringan.

Daftar Isi


- Siapkan instance Anda untuk jaringan yang ditingkatkan

- Menguji apakah jaringan yang ditingkatkan diaktifkan
- Mengaktifkan jaringan yang ditingkatkan pada instans Anda
- Memecahkan masalah konektivitas

## Siapkan instance Anda untuk jaringan yang ditingkatkan

Untuk mempersiapkan jaringan yang ditingkatkan menggunakan antarmuka Intel 82599 VF, siapkan instans Anda sebagai berikut:

- Pastikan jenis instans adalah salah satu dari berikut ini: C3, C4, D2, I2, M4 (tidak termasuk`m4.16xlarge`), dan R3.
- Pastikan instans tersebut memiliki konektivitas internet.
- Jika memiliki data penting pada instans yang ingin Anda pertahankan, Anda harus melakukan back up data tersebut sekarang dengan membuat AMI dari instans Anda. Memperbarui kernel dan modul kernel, serta mengaktifkan atribut `sriovNetSupport`, dapat menyebabkan instans yang tidak kompatibel atau sistem operasi tidak dapat dijangkau. Jika Anda memiliki back up terbaru, data Anda akan tetap disimpan jika hal ini terjadi.
- Instans Linux — Luncurkan instance dari AMI HVM menggunakan kernel Linux versi 2.6.32 atau yang lebih baru. AMI Amazon Linux HVM terbaru memiliki modul yang diperlukan agar jaringan yang ditingkatkan diinstal dan memiliki atribut yang diperlukan ditetapkan. Oleh karena itu, jika Anda meluncurkan instans yang didukung Amazon EBS dan jaringan yang ditingkatkan menggunakan AMI HVM Amazon Linux saat ini, jaringan yang ditingkatkan telah diaktifkan untuk instans Anda.

  > **Warning**
  >
  > Jaringan yang ditingkatkan hanya didukung untuk instans HVM. Mengaktifkan jaringan yang ditingkatkan dengan instans PV bisa menjadikannya tidak dapat dijangkau. Mengatur atribut ini tanpa modul yang tepat atau versi modul juga bisa membuat instans Anda tidak dapat dijangkau.

- Instans Windows — Luncurkan instance dari AMI HVM 64-bit. Anda tidak dapat mengaktifkan jaringan yang disempurnakan pada Windows Server 2008. Jaringan yang ditingkatkan sudah diaktifkan untuk Windows Server 2012 R2 dan Windows Server 2016 dan AMI yang lebih baru. Windows Server 2012 R2 menyertakan driver Intel 1.0.15.3 dan kami sarankan Anda memutakhirkan driver itu ke versi terbaru menggunakan utilitas Pnputil.exe .

- Gunakan AWS CloudShell dari Konsol Manajemen AWS, atau instal dan konfigurasikan AWS CLI atau AWS Tools for Windows PowerShell pada komputer mana pun yang Anda pilih, sebaiknya desktop atau laptop lokal Anda. Untuk informasi selengkapnya, lihat Akses Amazon EC2 atau Panduan Pengguna AWS CloudShell. Jaringan yang ditingkatkan tidak dapat dikelola dari konsol Amazon EC2.

## Menguji apakah jaringan yang ditingkatkan diaktifkan

Verifikasi bahwa `sriovNetSupport` atribut disetel pada instance atau gambar.

AWS CLI

Untuk memeriksa atribut instance (sriovNetSupport)

Gunakan perintah berikut describe-instance-attribute. Jika atribut disetel, nilainya adalah`simple`.

```
aws ec2 describe-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --attribute sriovNetSupport
```

Untuk memeriksa atribut gambar (sriovNetSupport)

Gunakan perintah berikut describe-images. Jika atribut disetel, nilainya adalah`simple`.

```
aws ec2 describe-images \
    --image-id ami-0abcdef1234567890 \
    --query "Images[].SriovNetSupport"
```

PowerShell

Untuk memeriksa atribut instance (sriovNetSupport)

Gunakan Get-EC2InstanceAttribute cmdlet. Jika atribut disetel, nilainya adalah`simple`.

```
Get-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -Attribute sriovNetSupport
```

Untuk memeriksa atribut gambar (sriovNetSupport)

Gunakan perintah berikut describe-images. Jika atribut disetel, nilainya adalah`simple`.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).SriovNetSupport
```

## Mengaktifkan jaringan yang ditingkatkan pada instans Anda

Prosedur yang Anda gunakan tergantung pada sistem operasi instance.

> **Warning**
>
> Tidak ada cara untuk menonaktifkan atribut jaringan yang ditingkatkan setelah Anda mengaktifkannya.

### Amazon Linux

AMI Amazon Linux HVM terbaru memiliki modul `ixgbevf` yang diperlukan agar jaringan yang ditingkatkan diinstal dan memiliki set atribut `sriovNetSupport` yang diperlukan. Oleh karena itu, jika Anda meluncurkan tipe instans menggunakan AMI HVM Amazon Linux saat ini, jaringan yang ditingkatkan telah diaktifkan untuk instans Anda. Untuk informasi selengkapnya, lihat Menguji apakah jaringan yang ditingkatkan diaktifkan.

Jika Anda meluncurkan instans menggunakan AMI Amazon Linux versi lama dan jaringan yang ditingkatkan belum diaktifkan, gunakan prosedur berikut untuk mengaktifkan jaringan yang ditingkatkan.

Untuk mengaktifkan jaringan yang ditingkatkan

1. Terhubung ke instans Anda.
2. Dari instans, jalankan perintah berikut untuk memperbarui instans Anda dengan modul kernel dan kernel terbaru, termasuk `ixgbevf`:

   ```
   [ec2-user ~]$ sudo yum update
   ```

3. Dari komputer lokal Anda, nyalakan ulang instans Anda menggunakan konsol Amazon EC2 atau salah satu perintah berikut: reboot-instances (AWS CLI) atau Restart-EC2Instance (AWS Tools for Windows PowerShell).
4. Hubungkan lagi ke instans Anda dan verifikasi bahwa modul `ixgbevf` telah diinstal dan pada versi minimum yang disarankan menggunakan perintah modinfo ixgbevf dari Menguji apakah jaringan yang ditingkatkan diaktifkan.

5. [EBS-backed instance] Dari komputer lokal Anda, hentikan instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: stop-instances (AWS CLI) atau Stop-EC2Instance (AWS Tools for Windows PowerShell).

   [Instans yang didukung penyimpanan instans] Anda tidak dapat menghentikan instans untuk memodifikasi atribut. Sebaliknya, lewati ke prosedur berikutnya.

6. Dari komputer lokal Anda, aktifkan atribut jaringan yang ditingkatkan menggunakan salah satu dari perintah berikut ini:

   AWS CLI

   Gunakan perintah modify-instance-attribute sebagai berikut.

   ```
   aws ec2 modify-instance-attribute \
       --instance-id i-1234567890abcdef0 \
       --sriov-net-support simple
   ```

   PowerShell

   Gunakan Edit-EC2InstanceAttribute cmdlet sebagai berikut.

   ```
   Edit-EC2InstanceAttribute `
       -InstanceId i-1234567890abcdef0 `
       -SriovNetSupport "simple"
   ```

7. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di Buat EBS-backed AMI Amazon. AMI mewarisi atribut jaringan yang ditingkatkan dari instans. Oleh karena itu, Anda dapat menggunakan AMI ini untuk meluncurkan instans lain dengan jaringan yang ditingkatkan diaktifkan secara default.

8. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: start-instances (AWS CLI) atau Start-EC2Instance (AWS Tools for Windows PowerShell).

9. Hubungkan ke instans Anda dan verifikasi bahwa modul `ixgbevf` diinstal dan dimuat di antarmuka jaringan Anda menggunakan perintah ethtool -i eth***n*** dari Menguji apakah jaringan yang ditingkatkan diaktifkan.

Untuk mengaktifkan jaringan yang ditingkatkan (instans yang didukung penyimpanan instans)

Ikuti prosedur sebelumnya hingga langkah tempat Anda menghentikan instans. Buat AMI baru seperti yang dijelaskan di Buat S3-backed AMI Amazon, pastikan untuk mengaktifkan atribut jaringan yang ditingkatkan saat Anda mendaftarkan AMI.

AWS CLI

Gunakan perintah register-image sebagai berikut.

```
aws ec2 register-image --sriov-net-support simple ...
```

PowerShell

Gunakan Register-EC2Image sebagai berikut.

```
Register-EC2Image -SriovNetSupport "simple" ...
```

Ubuntu

Sebelum Anda mulai, periksa apakah jaringan yang ditingkatkan telah diaktifkan pada instans Anda.

Quick Start Ubuntu HVM AMI menyertakan driver yang diperlukan untuk jaringan yang ditingkatkan. Jika Anda memiliki versi `ixgbevf` yang lebih lama dari 2.16.4, Anda dapat menginstal paket kernel `linux-aws` untuk mendapatkan driver jaringan terbaru yang ditingkatkan.

Prosedur berikut menyediakan langkah-langkah umum untuk mengompilasi modul `ixgbevf` pada instans Ubuntu.

Untuk menginstal paket kernel **`linux-aws`**

1. Hubungkan ke instans Anda.
2. Perbarui cache paket dan paket.

```
ubuntu:~$ sudo apt-get update && sudo apt-get upgrade -y linux-aws
```

> **Important**
>
> Jika selama proses update Anda diminta untuk menginstal `grub`, gunakan `/dev/xvda` untuk menginstal `grub`, lalu pilih untuk mempertahankan versi `/boot/grub/menu.lst` saat ini.

### Distribusi Linux lainnya

Sebelum Anda mulai, periksa apakah jaringan yang ditingkatkan telah diaktifkan pada instans Anda. Quick Start HVM AMI terbaru menyertakan driver yang diperlukan untuk jaringan yang ditingkatkan, oleh karena itu Anda tidak perlu melakukan langkah tambahan.

Prosedur berikut menyediakan langkah-langkah umum jika Anda perlu mengaktifkan jaringan yang ditingkatkan dengan antarmuka Intel 82599 VF pada distribusi Linux selain Amazon Linux atau Ubuntu. Untuk informasi selengkapnya, seperti detail sintaksis untuk perintah, lokasi file, atau paket dan dukungan alat, lihat dokumentasi khusus untuk distribusi Linux Anda.

Untuk mengaktifkan jaringan yang ditingkatkan di Linux

1. Terhubung ke instans Anda.
2. Unduh sumber untuk `ixgbevf` modul pada instance Anda dari Sourceforge di https://sourceforge.net/projects/e1000/files/ixgbevf %20stable/.

   Versi `ixgbevf` yang lebih lama dari 2.16.4, termasuk versi 2.14.2, tidak dibuat dengan benar pada beberapa distribusi Linux, termasuk versi Ubuntu tertentu.
3. Lakukan kompilasi dan instal modul `ixgbevf` pada instans Anda.

   > **Warning**
   >
   > Jika Anda mengompilasi modul `ixgbevf` untuk kernel Anda saat ini lalu mengupgrade kernel Anda tanpa membuat kembali driver untuk kernel baru, sistem Anda mungkin akan kembali ke modul `ixgbevf` khusus distribusi pada boot ulang berikutnya. Ini dapat membuat sistem Anda tidak dapat dijangkau jika versi khusus distribusi tidak kompatibel dengan jaringan yang ditingkatkan.
4. Jalankan perintah sudo depmod untuk memperbarui dependensi modul.
5. Perbarui `initramfs` pada instans Anda untuk memastikan bahwa modul baru dimuat pada saat boot.
6. Tentukan apakah sistem Anda menggunakan nama antarmuka jaringan yang dapat diprediksi secara default. Sistem yang menggunakan systemd atau udev versi 197 atau lebih tinggi dapat mengganti nama perangkat Ethernet dan tidak menjamin bahwa satu antarmuka jaringan akan dinamai `eth0`. Perilaku ini dapat menyebabkan masalah saat terhubung ke instans Anda. Untuk informasi lebih lanjut dan untuk melihat opsi konfigurasi lainnya, lihat Nama Antarmuka Jaringan yang Dapat Diprediksi di situs web freedesktop.org.

a. Anda dapat memeriksa udev versi systemd atau pada RPM-based sistem dengan perintah berikut:

```
[ec2-user ~]$ rpm -qa | grep -e '^systemd-[0-9]\+\|^udev-[0-9]\+'
    systemd-208-11.el7_0.2.x86_64
```

Dalam contoh Red Hat Enterprise Linux 7 di atas, versi systemd adalah 208, jadi nama antarmuka jaringan yang dapat diprediksi harus dinonaktifkan.

b. Nonaktifkan nama antarmuka jaringan yang dapat diprediksi dengan menambahkan opsi `net.ifnames=0` ke baris `GRUB_CMDLINE_LINUX` di `/etc/default/grub`.

```
[ec2-user ~]$ sudo sed -i '/^GRUB\_CMDLINE\_LINUX/s/\"$/\ net\.ifnames\=0\"/' /
etc/default/grub
```

c. Buat ulang file konfigurasi grub.

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

7. [EBS-backed instance] Dari komputer lokal Anda, hentikan instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: stop-instance (AWS CLI) atau Stop-EC2Instance ().AWS Tools for Windows PowerShell

   [Instans yang didukung penyimpanan instans] Anda tidak dapat menghentikan instans untuk memodifikasi atribut. Sebaliknya, lewati ke prosedur berikutnya.

8. Dari komputer lokal Anda, aktifkan atribut jaringan yang ditingkatkan menggunakan salah satu dari perintah berikut ini:

   AWS CLI

   Gunakan perintah modify-instance-attribute sebagai berikut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 -\
    -sriov-net-support simple
```

   PowerShell

   Gunakan Edit-EC2InstanceAttribute cmdlet sebagai berikut.

```
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -SriovNetSupport "simple"
```

9. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di Buat EBS-backed AMI Amazon. AMI mewarisi atribut jaringan yang ditingkatkan dari instans. Oleh karena itu, Anda dapat menggunakan AMI ini untuk meluncurkan instans lain dengan jaringan yang ditingkatkan diaktifkan secara default.

   Jika sistem operasi instans Anda berisi file `/etc/udev/rules.d/70-persistent-net.rules`, Anda harus menghapusnya sebelum membuat AMI. File ini berisi alamat MAC untuk adaptor Ethernet dari instans asli. Jika instans lain melakukan booting dengan file ini, sistem operasi tersebut `eth0` tidak akan dapat menemukan perangkat dan mungkin gagal, yang menyebabkan masalah booting. File ini dibuat ulang pada siklus boot berikutnya, dan setiap instans yang diluncurkan dari AMI membuat versi file mereka sendiri.
10. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: start-instances (AWS CLI) atau Start-EC2Instance (AWS Tools for Windows PowerShell).
11. (Opsional) Hubungkan ke instans Anda dan verifikasi bahwa modul telah diinstal.

Untuk mengaktifkan jaringan yang ditingkatkan (instans yang didukung penyimpanan instans)

Ikuti prosedur sebelumnya hingga langkah tempat Anda menghentikan instans. Buat AMI baru seperti yang dijelaskan di Buat S3-backed AMI Amazon, pastikan untuk mengaktifkan atribut jaringan yang ditingkatkan saat Anda mendaftarkan AMI.

AWS CLI

Gunakan perintah register-image sebagai berikut.

```
aws ec2 register-image --sriov-net-support simple ...
```

PowerShell

Gunakan Register-EC2Image sebagai berikut.

```
Register-EC2Image -SriovNetSupport "simple" ...
```

Windows

Jika Anda meluncurkan instans dan instans tersebut belum mengaktifkan jaringan yang ditingkatkan, Anda harus mengunduh dan menginstal driver adaptor jaringan yang diperlukan pada instans Anda, lalu menyetel atribut instans `sriovNetSupport` untuk mengaktifkan jaringan yang ditingkatkan. Anda hanya dapat mengaktifkan atribut ini pada tipe instans yang didukung. Untuk informasi selengkapnya, lihat Jaringan yang disempurnakan di instans Amazon EC2.

> **Important**
>
> Untuk melihat pembaruan driver terbaru di AMI Windows, lihat ri wayat versi AMI Windows di Refer AWS ensi AMI Windows.

Untuk mengaktifkan jaringan yang ditingkatkan

1. Hubungkan ke instans Anda dan masuk sebagai administrator lokal.
2. [Windows Server 2016 dan yang lebih baru] Jalankan PowerShell skrip Peluncuran EC2 berikut untuk mengonfigurasi instance setelah driver diinstal.

   ```
   PS C:\> C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeInstance.ps1 -Schedule
   ```

   > **Important**
   >
   > Kata sandi administrator akan diatur ulang ketika Anda mengaktifkan skrip EC2Launch inisialisasi instans. Anda dapat memodifikasi file konfigurasi untuk menonaktifkan pengaturan ulang kata sandi administrator dengan menentukannya di pengaturan untuk tugas inisialisasi.

3. Dari instans, unduh driver adaptor jaringan Intel untuk sistem operasi Anda:

   - Windows Server 2022

     Kunjungi halaman unduh dan unduh `Wired_driver_`*`version`*`_x64.zip`.
   - Windows Server 2019 termasuk untuk Server versi 1809 dan yang lebih baru*

     Kunjungi halaman unduh dan unduh `Wired_driver_`*`version`*`_x64.zip`.
   - Windows Server 2016 termasuk untuk Server versi 1803 dan sebelumnya*

Kunjungi halaman unduh dan unduh `Wired_driver_version_x64.zip`.

- Windows Server 2012 R2

  Kunjungi halaman unduh dan unduh `Wired_driver_version_x64.zip`.

- Windows Server 2012

  Kunjungi halaman unduh dan unduh `Wired_driver_version_x64.zip`.

- Windows Server 2008 R2

  Kunjungi halaman unduh dan unduh `PROWinx64Legacy.exe`.

*Server versi 1803 dan sebelumnya serta 1809 dan yang lebih baru tidak secara khusus ditujukan pada halaman Driver dan Software Intel.

4. Instal driver adaptor jaringan Intel untuk sistem operasi Anda.

   - Windows Server 2008 R2

     1. Di folder Unduh, cari file `PROWinx64Legacy.exe` dan namakan `PROWinx64Legacy.zip`.
     2. Ekstrak isi file `PROWinx64Legacy.zip` tersebut.
     3. Buka baris perintah, navigasi ke folder yang diekstrak, dan jalankan perintah berikut untuk menggunakan utilitas `pnputil` untuk menambahkan dan menginstal file INF di penyimpanan driver.

        ```
        C:\> pnputil -a PROXGB\Winx64\NDIS62\vxn62x64.inf
        ```

   - Windows Server 2022, Windows Server 2019, Windows Server 2016, Windows Server 2012 R2, dan Windows Server 2012

     1. Di folder Unduhan, ekstrak isi file `Wired_driver_version_x64.zip` tersebut.
     2. Buka baris perintah, navigasi ke folder yang diekstrak, dan jalankan perintah berikut untuk menggunakan utilitas `pnputil` untuk menambahkan dan menginstal file INF di penyimpanan driver.

        - Windows Server 2022

          ```
          pnputil -i -a PROXGB\Winx64\NDIS68\vxn68x64.inf
          ```

        - Windows Server 2019

```
pnputil -i -a PROXGB\Winx64\NDIS68\vxn68x64.inf
```

- Windows Server 2016

```
pnputil -i -a PROXGB\Winx64\NDIS65\vxn65x64.inf
```

- Windows Server 2012 R2

```
pnputil -i -a PROXGB\Winx64\NDIS64\vxn64x64.inf
```

- Windows Server 2012

```
pnputil -i -a PROXGB\Winx64\NDIS63\vxn63x64.inf
```

5. Dari komputer lokal Anda, aktifkan atribut jaringan yang ditingkatkan menggunakan salah satu dari perintah berikut ini:

   AWS CLI

   Gunakan perintah modify-instance-attribute sebagai berikut.

   ```
   aws ec2 modify-instance-attribute \
       --instance-id i-1234567890abcdef0 \
       --sriov-net-support simple
   ```

   PowerShell

   Gunakan Edit-EC2InstanceAttribute cmdlet sebagai berikut.

   ```
   Edit-EC2InstanceAttribute `
       -InstanceId i-1234567890abcdef0 `
       -SriovNetSupport "simple"
   ```

6. (Opsional) Buat AMI dari instans, seperti yang dijelaskan di Buat EBS-backed AMI Amazon. AMI mewarisi atribut jaringan yang ditingkatkan dari instans. Oleh karena itu, Anda dapat menggunakan AMI ini untuk meluncurkan instans lain dengan jaringan yang ditingkatkan diaktifkan secara default.
7. Dari komputer lokal Anda, mulai instance menggunakan konsol Amazon EC2 atau salah satu perintah berikut: start-instances (AWS CLI) atau Start-EC2Instance (AWS Tools for Windows PowerShell).

### Memecahkan masalah konektivitas

Jika Anda kehilangan konektivitas saat mengaktifkan jaringan yang ditingkatkan, modul `ixgbevf` mungkin tidak kompatibel dengan kernel tersebut. Coba instal versi modul `ixgbevf` yang disertakan dengan distribusi Linux untuk instans Anda.

Jika Anda mengaktifkan jaringan yang ditingkatkan untuk instans PV atau AMI, ini dapat membuat instans Anda tidak dapat dijangkau.

Untuk informasi selengkapnya, lihat Bagaimana cara mengaktifkan dan mengonfigurasi jaringan yang disempurnakan pada instans EC2 saya?

## Memantau kinerja jaringan untuk pengaturan ENA pada instans EC2 Anda

Driver Adaptor Jaringan Elastis (ENA) menerbitkan metrik performa jaringan dari instans di mana mereka diaktifkan. Anda dapat menggunakan metrik ini untuk memecahkan masalah performa instans, memilih ukuran instans yang tepat untuk beban kerja, rencana aktivitas penskalaan secara proaktif, dan aplikasi tolok ukur untuk menentukan apakah mereka memaksimalkan performa yang tersedia pada instans.

Amazon EC2 mendefinisikan maksimum jaringan pada tingkat instans untuk memastikan pengalaman jaringan berkualitas tinggi, termasuk kinerja jaringan yang konsisten di seluruh ukuran instans. AWS memberikan maksimum untuk yang berikut untuk setiap contoh:

- Kemampuan bandwidth — Setiap instans EC2 memiliki bandwidth maksimum untuk mengumpulkan lalu lintas masuk dan keluar, berdasarkan tipe dan ukuran instans. Beberapa contoh menggunakan mekanisme I/O kredit jaringan untuk mengalokasikan bandwidth jaringan berdasarkan pemanfaatan bandwidth rata-rata. Amazon EC2 juga memiliki bandwidth maksimum untuk lalu lintas ke Direct Connect dan internet. Untuk informasi selengkapnya, lihat Bandwidth jaringan instans Amazon EC2.
- Packet-per-second Kinerja (PPS) — Setiap instans EC2 memiliki kinerja PPS maksimum, berdasarkan jenis dan ukuran instans.
- Koneksi dilacak — Grup keamanan melacak setiap sambungan yang dibuat untuk memastikan bahwa paket kembali dikirim seperti yang diharapkan. Ada jumlah maksimum koneksi yang dapat dilacak per instans. Untuk informasi selengkapnya, lihat Pelacakan koneksi grup keamanan Amazon EC2

- Link-local akses layanan — Amazon EC2 menyediakan PPS maksimum per antarmuka jaringan untuk lalu lintas ke layanan proxy lokal seperti layanan Amazon DNS, Layanan Metadata Instance, dan Layanan Sinkronisasi Waktu Amazon.

Ketika lalu lintas jaringan untuk sebuah instance melebihi maksimum, membentuk AWS lalu lintas yang melebihi maksimum dengan mengantri dan kemudian menjatuhkan paket jaringan. Anda dapat memantau kapan lalu lintas melebihi maksimum menggunakan metrik performa jaringan. Metrik ini memberi tahu Anda, secara langsung, tentang dampak terhadap lalu lintas jaringan dan kemungkinan masalah performa jaringan.

Daftar Isi


- Persyaratan
- Metrik untuk driver ENA
- Melihat metrik performa jaringan untuk instans Anda
- Metrik untuk ENA Ekspres
- Metrik performa jaringan dengan driver DPDK untuk ENA
- Metrik pada instans yang berjalan FreeBSD


## Persyaratan

Instans Linux

- Menginstal driver ENA versi 2.2.10 atau yang lebih baru. Untuk memverifikasi versi terinstal, gunakan perintah ethtool berikut. Dalam contoh berikut, versi memenuhi persyaratan minimum.

```
[ec2-user ~]$ ethtool -i eth0 | grep version
version: 2.2.10
```

  Untuk memperbarui driver ENA Anda, lihat Jaringan yang ditingkatkan.
- Untuk mengimpor metrik ini ke Amazon CloudWatch, instal CloudWatch agen. Untuk informasi selengkapnya, lihat Mengumpulkan metrik kinerja jaringan di Panduan CloudWatch Pengguna Amazon.
- Untuk mendukung `conntrack_allowance_available` metrik, instal driver ENA versi 2.8.1 atau yang lebih baru.

- Untuk mengganti batas PPS fragmen keluar 1024, instal driver ENA versi 2.13.3 atau yang lebih baru.

Instans Windows

- Menginstal driver ENA versi 2.2.2 atau yang lebih baru. Untuk memverifikasi versi yang diinstal, gunakan Pengelola Perangkat sebagai berikut.
  1. Buka Pengelola Perangkat dengan menjalankan `devmgmt.msc`.
  2. Perluas Adaptor Jaringan.
  3. Pilih Amazon Elastic Network Adapter, Properti.
  4. Pada tab Driver, temukan Versi Driver.

  Untuk memperbarui driver ENA Anda, lihat Jaringan yang ditingkatkan.
- Untuk mengimpor metrik ini ke Amazon CloudWatch, instal CloudWatch agen. Untuk informasi selengkapnya, lihat Mengumpulkan metrik jaringan lanjutan di Panduan CloudWatch Pengguna Amazon.

## Metrik untuk driver ENA

Driver ENA memberikan metrik berikut untuk instans secara langsung. Mereka menyediakan jumlah kumulatif paket antri atau dijatuhkan pada setiap antarmuka jaringan sejak driver terakhir diatur ulang.

| Metrik | Deskripsi | Didukung pada |
| --- | --- | --- |
| `bw_in_allowance_exceeded` | Jumlah paket antri atau dijatuhkan karena kumpulan bandwidth yang masuk melebihi maksimum untuk instans. | Semua tipe instans |
| `bw_out_allowance_exceeded` | Jumlah paket antre atau dijatuhkan karena bandwidth agregat yang keluar melebihi maksimum untuk instans. | Semua tipe instans |
| `conntrack_allowance_exceeded` | Jumlah paket turun karena pelacakan koneksi melebihi | Semua tipe instans |

| Metrik | Deskripsi | Didukung pada |
| --- | --- | --- |
| | maksimum untuk instans dan koneksi baru tidak dapat dibuat. Hal ini dapat mengakibatkan hilangnya paket untuk lalu lintas ke atau dari instans. | |
| `conntrack_allowance_available` | Jumlah koneksi yang dilacak yang dapat dibuat oleh instans sebelum menekan tunjangan Connections Tracked dari tipe instans tersebut. | Nitro-based contoh saja |
| `linklocal_allowance_exceeded` | Jumlah paket turun karena PPS lalu lintas ke layanan proksi lokal melebihi batas maksimum untuk antarmuka jaringan. Ini berdampak pada lalu lintas ke layanan Amazon DNS, Layanan Metadata Instans, dan Layanan Sinkronisasi Waktu Amazon, tetapi tidak memengaruhi lalu lintas ke resolver DNS khusus. | Semua tipe instans |
| `pps_allowance_exceeded` | Jumlah paket yang mengantri atau dijatuhkan karena PPS dua arah melebihi maksimum untuk instance. * | Semua tipe instans |

* Tergantung pada pengaturan mode proxy fragmen untuk driver ENA Linux v2.13.3 atau yang lebih baru, batas ini juga dapat mencakup penurunan fragmen keluar yang melebihi 1024 PPS untuk antarmuka jaringan. Jika mode proxy fragmen diaktifkan untuk driver Linux, penurunan fragmen keluar melewati batas 1024 PPS yang biasanya berlaku dan dihitung dalam tunjangan PPS standar. Mode proxy fragmen dinonaktifkan secara default.

## Melihat metrik performa jaringan untuk instans Anda

Prosedur yang Anda gunakan tergantung pada sistem operasi instance.

### Instans Linux

Anda dapat menerbitkan metrik ke alat favorit Anda untuk memvisualisasikan data metrik. Misalnya, Anda dapat mempublikasikan metrik ke Amazon CloudWatch menggunakan CloudWatch agen. Agen memungkinkan Anda untuk memilih metrik individu dan mengendalikan publikasi.

Anda juga dapat menggunakan ethtool untuk mengambil metrik untuk setiap antarmuka jaringan, seperti eth0, sebagai berikut.

```
[ec2-user ~]$ ethtool -S eth0
     bw_in_allowance_exceeded: 0
     bw_out_allowance_exceeded: 0
     pps_allowance_exceeded: 0
     conntrack_allowance_exceeded: 0
     linklocal_allowance_exceeded: 0
     conntrack_allowance_available: 136812
```

### Instans Windows

Anda dapat melihat metrik menggunakan pengukur performa Windows. Data dapat diurai sesuai dengan EnaPerfCounters manifes. Ini adalah file XML yang menentukan penyedia pengukur performa dan rangkaian penghitungnya.

#### Untuk menginstal manifes

Jika Anda meluncurkan instans menggunakan AMI yang berisi driver ENA 2.2.2 atau yang lebih baru, atau menggunakan skrip instal dalam paket driver ENA 2.2.2, manifes sudah terinstal. Untuk menginstal manifes secara manual, gunakan langkah-langkah berikut:

1. Menghapus manifes yang ada menggunakan perintah berikut:

   ```
   unlodctr /m:EnaPerfCounters.man
   ```

2. Salin file manifes `EnaPerfCounters.man` dari paket instalasi driver ke `%SystemRoot%\System32\drivers`.
3. Instal manifes baru menggunakan perintah berikut:

```
lodctr /m:EnaPerfCounters.man
```

Untuk melihat metrik menggunakan Performance Monitor

1. Buka Monitor Performa.
2. Tekan Ctrl+N untuk menambahkan penghitung baru.
3. Pilih ENA Packets Shaping dari daftar.
4. Pilih instans untuk memantau dan pilih Tambahkan.
5. Pilih OKE.

## Metrik untuk ENA Ekspres

ENA Express didukung oleh teknologi AWS Scalable Reliable Datagram (SRD). SRD adalah protokol transportasi jaringan performa tinggi yang menggunakan perutean dinamis untuk meningkatkan throughput dan meminimalkan latensi ekor. Jika Anda telah mengaktifkan ENA Express untuk lampiran antarmuka jaringan pada instans pengirim dan instans penerima, Anda dapat menggunakan metrik ENA Express untuk membantu memastikan bahwa instans Anda memanfaatkan sepenuhnya peningkatan kinerja yang disediakan teknologi SRD. Contoh:

- Evaluasi sumber daya Anda untuk memastikan bahwa mereka memiliki kapasitas yang cukup untuk membangun lebih banyak koneksi SRD.
- Identifikasi di mana ada potensi masalah yang mencegah paket keluar yang memenuhi syarat menggunakan SRD.
- Hitung persentase lalu lintas keluar yang menggunakan SRD untuk instans tersebut.
- Hitung persentase lalu lintas masuk yang menggunakan SRD untuk instans tersebut.

> **Note**
>
> Untuk menghasilkan metrik, gunakan driver versi 2.8 atau lebih tinggi.

Untuk melihat daftar metrik untuk instance Linux Anda yang difilter untuk ENA Express, jalankan ethtool perintah berikut untuk antarmuka jaringan Anda (ditampilkan di sini sebagai`eth0`). Perhatikan nilai met `ena_srd_mode` rik.

```
[ec2-user ~]$ ethtool -S eth0 | grep ena_srd
NIC statistics:
 ena_srd_mode: 1
 ena_srd_tx_pkts: 0
 ena_srd_eligible_tx_pkts: 0
 ena_srd_rx_pkts: 0
 ena_srd_resource_utilization: 0
```

Metrik berikut tersedia untuk semua instans yang mengaktifkan ENA Express.

ena_srd_mode

Menjelaskan fitur ENA Express mana yang diaktifkan. Nilai adalah sebagai berikut:

- `0` = ENA Ekspres mati, UDP mati
- `1` = ENA Ekspres aktif, UDP mati
- `2` = ENA Ekspres mati, UDP aktif

> **Note**
>
> Ini hanya terjadi ketika ENA Ekspres awalnya diaktifkan, dan UDP dikonfigurasi untuk menggunakannya. Nilai sebelumnya dipertahankan untuk lalu lintas UDP.

- `3` = ENA Ekspres aktif, UDP aktif

ena_srd_eligible_tx_pkts

Jumlah jaringan sebagai berikut:

- Baik tipe instans pengiriman maupun penerimaan didukung. Lihat tabel Tipe instans yang didukung untuk ENA Ekspres untuk informasi selengkapnya.
- Instans pengiriman dan penerimaan harus memiliki ENA Ekspres yang dikonfigurasi.
- Instans pengirim dan penerimaan harus berjalan di Wilayah yang sama.
- Jalur jaringan antara instans tidak boleh menyertakan kotak perangkat lunak perantara (middleware). ENA Ekspres saat ini tidak mendukung kotak perangkat lunak perantara (middleware).

> **Note**
>
> Metrik kelayakan ENA Ekspres mencakup persyaratan sumber dan tujuan, dan jaringan antara dua titik akhir. Paket yang memenuhi syarat masih dapat didiskualifikasi setelah

dihitung. Misalnya, jika paket yang memenuhi syarat melebihi batas unit transmisi maksimum (MTU), paket tersebut jatuh kembali ke transmisi ENA standar, meskipun paket tersebut masih tercermin sebagai memenuhi syarat di konter.

ena_srd_tx_pkts

Jumlah paket SRD yang dikirimkan dalam periode waktu tertentu.

ena_srd_rx_pkts

Jumlah paket SRD yang diterima dalam periode waktu tertentu.

ena_srd_pemanfaatan sumber daya

Persentase pemanfaatan memori maksimum yang diizinkan untuk koneksi SRD bersamaan yang telah dikonsumsi instans.

Untuk mengonfirmasi apakah transmisi paket menggunakan SRD, Anda dapat membandingkan jumlah paket yang memenuhi syarat (metrik `ena_srd_eligible_tx_pkts`) dengan jumlah paket SRD yang ditransmisikan (metrik `ena_srd_tx_pkts`) selama periode waktu tertentu.

Lalu lintas keluar (paket keluar)

Untuk memastikan bahwa lalu lintas jalan keluar Anda menggunakan SRD seperti yang diharapkan, bandingkan jumlah paket yang memenuhi syarat SRD (`ena_srd_eligible_tx_pkts`) dengan jumlah paket SRD yang dikirim (`ena_srd_tx_pkts`) selama periode waktu tertentu.

Perbedaan yang signifikan antara jumlah paket yang memenuhi syarat dan jumlah paket SRD yang dikirim sering disebabkan oleh masalah pemanfaatan sumber daya. Ketika kartu jaringan yang terpasang pada instance telah menghabiskan sumber daya maksimumnya, atau jika paket melebihi batas MTU, paket yang memenuhi syarat tidak dapat mengirimkan melalui SRD, dan harus kembali ke transmisi ENA standar. Paket juga dapat jatuh ke dalam celah ini selama migrasi langsung atau pembaruan server langsung. Pemecahan masalah tambahan diperlukan untuk menentukan akar penyebabnya.

> **Note**
>
> Anda dapat mengabaikan perbedaan kecil sesekali antara jumlah paket yang memenuhi syarat dan jumlah paket SRD. Ini dapat terjadi ketika instans Anda membuat koneksi ke instans lain untuk lalu lintas SRD, misalnya.

Untuk mengetahui berapa persentase total lalu lintas keluar Anda selama periode waktu tertentu menggunakan SRD, bandingkan jumlah paket SRD yang dikirim (`ena_srd_tx_pkts`) dengan jumlah total paket yang dikirim untuk instans (`NetworkPacketOut`) selama waktu itu.

Lalu lintas masuk (paket masuk)

Untuk mengetahui berapa persentase lalu lintas masuk Anda menggunakan SRD, bandingkan jumlah paket SRD yang diterima (`ena_srd_rx_pkts`) selama periode waktu tertentu dengan jumlah total paket yang diterima untuk instans (`NetworkPacketIn`) selama waktu itu.

Pemanfaatan Sumber Daya

Pemanfaatan sumber daya didasarkan pada jumlah koneksi SRD bersamaan yang dapat ditahan oleh satu instans pada waktu tertentu. Metrik pemanfaatan sumber daya (`ena_srd_resource_utilization`) melacak pemanfaatan Anda saat ini untuk instans tersebut. Saat pemanfaatan mendekati 100%, Anda dapat mengharapkan untuk melihat masalah performa. ENA Ekspres jatuh kembali dari SRD ke transmisi ENA standar, dan kemungkinan paket yang dijatuhkan meningkat. Pemanfaatan sumber daya yang tinggi adalah tanda bahwa sudah waktunya untuk meningkatkan skala instans untuk meningkatkan performa jaringan.

> **Note**
>
> Ketika lalu lintas jaringan untuk sebuah instance melebihi maksimum, membentuk AWS lalu lintas yang melebihi maksimum dengan mengantri dan kemudian menjatuhkan paket jaringan.

Tetap

Metrik jalan keluar dan masuknya bertambah saat ENA Ekspres diaktifkan untuk instans. Metrik berhenti bertambah jika ENA Ekspres dinonaktifkan, tetapi tetap ada selama instans masih berjalan. Metrik diatur ulang jika instans reboot atau diakhiri, atau jika antarmuka jaringan terlepas dari instans.

## Metrik performa jaringan dengan driver DPDK untuk ENA

ENA driver versi 2.2.0 dan yang lebih baru mendukung pelaporan metrik jaringan. DPDK 20.11 termasuk driver ENA 2.2.0 dan versi DPDK pertama untuk mendukung fitur ini.

Driver DPDK v25.03 atau yang lebih baru mendukung mode proxy fragmen. Jika mode proxy fragmen diaktifkan untuk driver DPDK, penurunan fragmen keluar melewati batas 1024 PPS yang biasanya berlaku dan dihitung dalam tunjangan PPS standar. Mode proxy fragmen dinonaktifkan secara default.

Anda dapat menggunakan contoh aplikasi untuk melihat statistik DPDK. Untuk memulai versi interaktif aplikasi contoh, jalankan perintah berikut.

```
./app/dpdk-testpmd -- -i
```

Dalam sesi interaktif ini, Anda dapat memasukkan perintah untuk mengambil statistik diperpanjang untuk port. Contoh perintah berikut mengambil statistik untuk port 0.

```
show port xstats 0
```

Berikut ini adalah contoh sesi interaktif dengan aplikasi contoh DPDK.

```
[root@ip-192.0.2.0 build]# ./app/dpdk-testpmd -- -i
        EAL: Detected 4 lcore(s)
        EAL: Detected 1 NUMA nodes
        EAL: Multi-process socket /var/run/dpdk/rte/mp_socket
        EAL: Selected IOVA mode 'PA'
        EAL: Probing VFIO support...
        EAL:   Invalid NUMA socket, default to 0
        EAL:   Invalid NUMA socket, default to 0
        EAL: Probe PCI driver: net_ena (1d0f:ec20) device: 0000:00:06.0
(socket 0)
        EAL: No legacy callbacks, legacy socket not created
        Interactive-mode selected

        Port 0: link state change event
        testpmd: create a new mbuf pool <mb_pool_0>: n=171456,
size=2176, socket=0
        testpmd: preferred mempool ops selected: ring_mp_mc

        Warning! port-topology=paired and odd forward ports number, the
last port will pair with itself.
```

```
        Configuring Port 0 (socket 0)
        Port 0: 02:C7:17:A2:60:B1
        Checking link statuses...
        Done
        Error during enabling promiscuous mode for port 0: Operation
not supported - ignore
        testpmd> show port xstats 0
        ###### NIC extended statistics for port 0
        rx_good_packets: 0
        tx_good_packets: 0
        rx_good_bytes: 0
        tx_good_bytes: 0
        rx_missed_errors: 0
        rx_errors: 0
        tx_errors: 0
        rx_mbuf_allocation_errors: 0
        rx_q0_packets: 0
        rx_q0_bytes: 0
        rx_q0_errors: 0
        tx_q0_packets: 0
        tx_q0_bytes: 0
        wd_expired: 0
        dev_start: 1
        dev_stop: 0
        tx_drops: 0
        bw_in_allowance_exceeded: 0
        bw_out_allowance_exceeded: 0
        pps_allowance_exceeded: 0
        conntrack_allowance_exceeded: 0
        linklocal_allowance_exceeded: 0
        rx_q0_cnt: 0
        rx_q0_bytes: 0
        rx_q0_refill_partial: 0
        rx_q0_bad_csum: 0
        rx_q0_mbuf_alloc_fail: 0
        rx_q0_bad_desc_num: 0
        rx_q0_bad_req_id: 0
        tx_q0_cnt: 0
        tx_q0_bytes: 0
        tx_q0_prepare_ctx_err: 0
        tx_q0_linearize: 0
        tx_q0_linearize_failed: 0
        tx_q0_tx_poll: 0
```

```
        tx_q0_doorbells: 0
        tx_q0_bad_req_id: 0
        tx_q0_available_desc: 1023
        testpmd>
```

Untuk informasi lebih lanjut tentang aplikasi contoh dan menggunakannya untuk mengambil statistik yang diperluas. lihat Panduan Pengguna Aplikasi Testpmd dalam dokumentasi DPDK.

### Metrik pada instans yang berjalan FreeBSD

Dimulai dengan versi 2.3.0, FreeBSD driver ENA mendukung pengumpulan metrik kinerja jaringan pada instans yang berjalanFreeBSD. Untuk mengaktifkan pengumpulan FreeBSD metrik, masukkan perintah berikut dan atur *interval* ke nilai antara 1 dan 3600. Ini menentukan seberapa sering, dalam hitungan detik, untuk mengumpulkan FreeBSD metrik.

```
sysctl dev.ena.network_interface.eni_metrics.sample_interval=interval
```

Misalnya, perintah berikut mengatur driver untuk mengumpulkan FreeBSD metrik pada antarmuka jaringan 1 setiap 10 detik:

```
sysctl dev.ena.1.eni_metrics.sample_interval=10
```

Untuk mematikan kumpulan FreeBSD metrik, Anda dapat menjalankan perintah sebelumnya dan menentukan `0` sebagai*interval*.

Setelah mengaktifkan pengumpulan FreeBSD metrik, Anda dapat mengambil kumpulan metrik yang dikumpulkan terbaru dengan menjalankan perintah berikut.

```
sysctl dev.ena.network_interface.eni_metrics
```

## Meningkatkan latensi jaringan untuk instans EC2 berbasis Linux

Latensi jaringan adalah jumlah waktu yang dibutuhkan untuk paket data untuk melakukan perjalanan dari sumbernya ke tujuannya. Aplikasi yang mengirim data ke seluruh jaringan bergantung pada respons tepat waktu untuk memberikan pengalaman pengguna yang positif. Latensi jaringan yang tinggi dapat menyebabkan berbagai masalah, seperti berikut ini:

- Waktu muat lambat untuk halaman web
- Kelambatan streaming video

- Kesulitan mengakses sumber daya online

Bagian ini menguraikan langkah-langkah yang dapat Anda ambil untuk meningkatkan latensi jaringan pada instans Amazon EC2 yang berjalan di Linux. Untuk mencapai latensi optimal, ikuti langkah-langkah berikut untuk mengonfigurasi pengaturan driver instans, kernel, dan ENA Anda. Untuk panduan konfigurasi tambahan, lihat Panduan Prakti k Terbaik Driver Linux ENA dan Optimalisasi Kinerja diGitHub.

> **Note**
>
> Langkah dan pengaturan mungkin sedikit berbeda, tergantung pada perangkat keras jaringan spesifik Anda, AMI tempat Anda meluncurkan instans, dan kasus penggunaan aplikasi Anda. Sebelum Anda membuat perubahan apa pun, uji dan pantau performa jaringan Anda secara menyeluruh untuk memastikan bahwa Anda mendapatkan hasil yang diinginkan.

## Mengurangi jumlah hop jaringan untuk paket data

Setiap lompatan yang diambil paket data saat bergerak dari router ke router meningkatkan latensi jaringan. Biasanya, lalu lintas harus mengambil beberapa lompatan untuk mencapai tujuan Anda. Ada dua cara untuk mengurangi lompatan jaringan untuk instans Amazon EC2 Anda, sebagai berikut:

- Grup penempatan klaster — Saat Anda menentukan grup penempatan klaster, Amazon EC2 meluncurkan instans yang berdekatan satu sama lain, secara fisik dalam Zona Ketersediaan (AZ) yang sama dengan kemasan yang lebih ketat. Kedekatan fisik instans dalam grup memungkinkan mereka untuk memanfaatkan konektivitas berkecepatan tinggi, menghasilkan latensi rendah dan throughput aliran tunggal yang tinggi.
- Host Khusus — Host Khusus adalah server fisik yang didedikasikan untuk Anda gunakan. Dengan Host Khusus, Anda dapat meluncurkan instans Anda untuk berjalan di server fisik yang sama. Komunikasi antar instans yang berjalan pada Host Khusus yang sama dapat terjadi tanpa lompatan jaringan tambahan.

## Bagaimana konfigurasi kernel Linux memengaruhi latensi

Konfigurasi kernel Linux dapat meningkatkan atau mengurangi latensi jaringan. Untuk mencapai tujuan optimasi latensi Anda, penting untuk menyempurnakan konfigurasi kernel Linux sesuai dengan persyaratan spesifik beban kerja Anda.

Ada banyak opsi konfigurasi untuk kernel Linux yang dapat membantu mengurangi latensi jaringan. Opsi yang paling berdampak adalah sebagai berikut.

- Aktifkan mode polling sibuk — Mode polling sibuk mengurangi latensi pada jalur penerimaan jaringan. Saat Anda mengaktifkan mode polling sibuk, kode lapisan soket dapat langsung melakukan polling antrean penerima perangkat jaringan. Kelemahan dari polling sibuk adalah penggunaan CPU yang lebih tinggi di host yang berasal dari polling untuk data baru dalam lingkaran ketat. Ada dua pengaturan global yang mengontrol jumlah mikrodetik untuk menunggu paket untuk semua antarmuka.

  `busy_read`

  Batas waktu polling sibuk latensi rendah untuk pembacaan soket. Ini mengontrol jumlah mikrodetik untuk menunggu lapisan soket membaca paket pada antrean perangkat. Untuk mengaktifkan fitur secara global dengan perintah sysctl, organisasi Kernel Linux merekomendasikan nilai 50 mikrodetik. Untuk informasi selengkapnya, lihat busy_read di panduan pengguna dan administrator kernel Linux.

  ```
  [ec2-user ~]$ sudo sysctl -w net.core.busy_read=50
  ```

  `busy_poll`

  Batas waktu polling sibuk latensi rendah untuk polling dan pilih. Ini mengontrol jumlah mikrodetik untuk menunggu peristiwa. Nilai yang disarankan adalah antara 50-100 mikrodetik, tergantung pada jumlah soket yang Anda polling. Semakin banyak soket yang Anda tambahkan, semakin tinggi angkanya.

  ```
  [ec2-user ~]$ sudo sysctl -w net.core.busy_poll=50
  ```

- Konfigurasikan status daya CPU (C-states) — C-states mengontrol tingkat tidur yang mungkin masuk inti saat tidak aktif. Anda mungkin ingin mengontrol C-states untuk menyetel sistem Anda untuk latensi versus kinerja. Lebih dalam C-states, CPU pada dasarnya “tidur” dan tidak dapat menanggapi permintaan sampai bangun dan beralih kembali ke keadaan aktif. Menidurkan inti membutuhkan waktu, dan meskipun inti tidur memungkinkan lebih banyak ruang kepala untuk inti lain untuk meningkat ke frekuensi yang lebih tinggi, inti tidur tersebut membutuhkan waktu untuk bangun kembali dan melakukan pekerjaan.

Misalnya, jika inti yang ditugaskan untuk menangani interupsi paket jaringan tertidur, mungkin ada penundaan dalam melayani interupsi tersebut. Anda dapat mengkonfigurasi sistem sehingga tidak menggunakan lebih dalam C-states. Namun, meskipun konfigurasi ini mengurangi latensi reaksi prosesor, konfigurasi ini juga mengurangi ruang kepala yang tersedia untuk inti lain untuk Turbo Boost.

Untuk mengurangi latensi reaksi prosesor, Anda dapat membatasi lebih dalam C-states. Untuk informasi selengkapnya, lihat Kinerja tinggi dan latensi rendah dengan membatasi lebih C-states dalam Panduan Pengguna Amazon Linux 2.

## Moderasi interupsi

Driver jaringan ENA memungkinkan komunikasi antara instans dan jaringan. Driver memproses paket jaringan dan meneruskannya ke tumpukan jaringan atau ke kartu Nitro. Ketika paket jaringan masuk, kartu Nitro menghasilkan interupsi bagi CPU untuk memberi tahu perangkat lunak suatu peristiwa.

Menginterupsi

Interupsi adalah sinyal yang dikirim perangkat atau aplikasi ke prosesor. Interupsi memberi tahu prosesor bahwa suatu peristiwa telah terjadi atau suatu kondisi telah terpenuhi yang membutuhkan perhatian segera. Interupsi dapat menangani tugas yang sensitif terhadap waktu seperti menerima data dari antarmuka jaringan, menangani peristiwa perangkat keras, atau melayani permintaan dari perangkat lain.

Moderasi interupsi

Moderasi interupsi adalah teknik yang mengurangi jumlah interupsi yang dihasilkan perangkat dengan menggabungkan atau menundanya. Tujuan dari moderasi interupsi adalah untuk meningkatkan performa sistem dengan mengurangi overhead yang terkait dengan penanganan sejumlah besar interupsi. Terlalu banyak interupsi meningkatkan penggunaan CPU, berdampak buruk pada throughput, sementara terlalu sedikit interupsi meningkatkan latensi.

Moderasi interupsi dinamis

Moderasi interupsi dinamis adalah bentuk peningkatan moderasi interupsi yang secara dinamis menyesuaikan tingkat interupsi berdasarkan beban sistem saat ini dan pola lalu lintas. Ini bertujuan untuk mencapai keseimbangan antara mengurangi interupsi overhead dan paket per detik, atau bandwidth.

> **Note**
>
> Moderasi interupsi dinamis diaktifkan secara default di beberapa AMI (tetapi dapat diaktifkan atau dinonaktifkan di semua AMI).

Untuk meminimalkan latensi jaringan, mungkin perlu menonaktifkan moderasi interupsi. Namun, ini juga dapat meningkatkan overhead pemrosesan interupsi. Penting untuk menemukan keseimbangan yang tepat antara mengurangi latensi dan meminimalkan overhead. Perintah `ethtool` dapat membantu Anda mengonfigurasi moderasi interupsi. Secara default, `rx-usecs` diatur ke `20`, dan `tx-usecs` diatur ke `64`.

Untuk mendapatkan konfigurasi modifikasi interupsi saat ini, gunakan perintah berikut.

```
[ec2-user ~]$ ethtool -c interface | egrep "rx-usecs:|tx-usecs:|Adaptive RX"
Adaptive RX: on  TX: off
rx-usecs: 20
tx-usecs: 64
```

Untuk menonaktifkan modifikasi interupsi dan moderasi interupsi dinamis, gunakan perintah berikut.

```
[ec2-user ~]$ sudo ethtool -C interface adaptive-rx off rx-usecs 0 tx-usecs 0
```

## Pertimbangan sistem nitro untuk penyetelan kinerja

Sistem Nitro adalah kumpulan komponen perangkat keras dan perangkat lunak yang dibangun oleh AWS yang memungkinkan kinerja tinggi, ketersediaan tinggi, dan keamanan tinggi. Sistem Nitro menyediakan kemampuan seperti bare metal yang menghilangkan overhead virtualisasi dan mendukung beban kerja yang memerlukan akses penuh ke perangkat keras host. Untuk informasi lebih rinci, lihat Sistem AWS Nitro.

Semua jenis instans EC2 generasi saat ini melakukan pemrosesan paket jaringan pada Kartu Nitro EC2. Topik ini mencakup penanganan paket tingkat tinggi pada kartu Nitro, aspek umum arsitektur jaringan dan konfigurasi yang memengaruhi kinerja penanganan paket, dan tindakan apa yang dapat Anda ambil untuk mencapai kinerja puncak untuk instans berbasis Nitro Anda.

Kartu Nitro menangani semua antarmuka input dan output (I/O), seperti yang diperlukan untuk Virtual Private Cloud (VPC). Untuk semua komponen yang mengirim atau menerima informasi melalui

jaringan, kartu Nitro bertindak sebagai perangkat komputasi mandiri untuk I/O lalu lintas yang secara fisik terpisah dari papan utama sistem tempat beban kerja pelanggan berjalan.

## Aliran paket jaringan pada kartu Nitro

Instans EC2 yang dibangun di atas sistem Nitro memiliki kemampuan akselerasi perangkat keras yang memungkinkan pemrosesan paket lebih cepat, yang diukur dengan kecepatan throughput paket per detik (PPS). Ketika kartu Nitro melakukan evaluasi awal untuk aliran baru, kartu ini menyimpan informasi yang sama untuk semua paket dalam aliran, seperti grup keamanan, daftar kontrol akses, dan entri tabel rute. Ketika memproses paket tambahan untuk aliran yang sama, ia dapat menggunakan informasi yang disimpan untuk mengurangi overhead untuk paket-paket tersebut.

Kecepatan koneksi Anda diukur dengan metrik koneksi per detik (CPS). Setiap koneksi baru memerlukan overhead pemrosesan tambahan yang harus diperhitungkan dalam perkiraan kemampuan beban kerja. Penting untuk mempertimbangkan metrik CPS dan PPS saat Anda merancang beban kerja Anda.

Bagaimana koneksi dibuat

Ketika koneksi dibuat antara instance berbasis Nitro dan titik akhir lain, kartu Nitro mengevaluasi aliran penuh untuk paket pertama yang dikirim atau diterima antara dua titik akhir. Untuk paket berikutnya dari aliran yang sama, evaluasi ulang penuh biasanya tidak diperlukan. Namun, ada pengecualian. Untuk informasi lebih lanjut tentang pengecualian, lihatPaket yang tidak menggunakan akselerasi perangkat keras.

Properti berikut menentukan dua titik akhir dan aliran paket di antara mereka. Kelima sifat ini bersama-sama dikenal sebagai aliran 5-tupel.

- IP sumber
- Port sumber
- IP Tujuan
- Port tujuan
- Protokol komunikasi

Arah aliran paket dikenal sebagai *input* (inbound) dan *egress* (outbound). Deskripsi tingkat tinggi berikut merangkum aliran paket jaringan ujung ke ujung.

- Ingress — Ketika kartu Nitro menangani paket jaringan masuk, ia mengevaluasi paket terhadap aturan firewall stateful dan daftar kontrol akses. Ini melacak koneksi, mengukurnya, dan melakukan tindakan lain sebagaimana berlaku. Kemudian meneruskan paket ke tujuannya pada CPU host.
- Egress — Ketika kartu Nitro menangani paket jaringan keluar, ia mencari tujuan antarmuka jarak jauh, mengevaluasi berbagai fungsi VPC, menerapkan batas kecepatan, dan melakukan tindakan lain yang berlaku. Kemudian meneruskan paket ke tujuan hop berikutnya di jaringan.

## Rancang jaringan Anda untuk kinerja optimal

Untuk memanfaatkan kemampuan kinerja sistem Nitro Anda, Anda harus memahami apa kebutuhan pemrosesan jaringan Anda dan bagaimana kebutuhan tersebut memengaruhi beban kerja untuk sumber daya Nitro Anda. Kemudian Anda dapat merancang untuk kinerja optimal untuk lanskap jaringan Anda. Pengaturan infrastruktur dan desain dan konfigurasi beban kerja aplikasi dapat memengaruhi pemrosesan paket dan kecepatan koneksi. Misalnya, jika aplikasi Anda memiliki tingkat pembentukan koneksi yang tinggi, seperti layanan DNS, firewall, atau router virtual, ia akan memiliki lebih sedikit kesempatan untuk memanfaatkan akselerasi perangkat keras yang hanya terjadi setelah koneksi dibuat.

Anda dapat mengonfigurasi pengaturan aplikasi dan infrastruktur untuk merampingkan beban kerja dan meningkatkan kinerja jaringan. Namun, tidak semua paket memenuhi syarat untuk akselerasi. Sistem Nitro menggunakan aliran jaringan penuh untuk koneksi baru dan untuk paket yang tidak memenuhi syarat untuk akselerasi.

Sisa bagian ini akan fokus pada pertimbangan desain aplikasi dan infrastruktur untuk membantu memastikan bahwa paket mengalir dalam jalur yang dipercepat sebanyak mungkin.

### Pertimbangan desain jaringan untuk sistem Nitro

Saat Anda mengkonfigurasi lalu lintas jaringan untuk instance Anda, ada banyak aspek yang perlu dipertimbangkan yang dapat memengaruhi kinerja PPS. Setelah aliran terbentuk, sebagian besar paket yang secara teratur masuk atau keluar memenuhi syarat untuk akselerasi. Namun, ada pengecualian untuk memastikan bahwa desain infrastruktur dan aliran paket terus memenuhi standar protokol.

Untuk mendapatkan kinerja terbaik dari kartu Nitro Anda, Anda harus mempertimbangkan dengan cermat pro dan kontra dari detail konfigurasi berikut untuk infrastruktur dan aplikasi Anda.

### Pertimbangan infrastruktur

Konfigurasi infrastruktur Anda dapat memengaruhi aliran paket dan efisiensi pemrosesan Anda. Daftar berikut mencakup beberapa pertimbangan penting.

#### Konfigurasi antarmuka jaringan dengan asimetri

Grup keamanan menggunakan pelacakan koneksi untuk melacak informasi tentang lalu lintas yang mengalir ke dan dari instance. Perutean asimetris, di mana lalu lintas masuk ke instance melalui satu antarmuka jaringan dan keluar melalui antarmuka jaringan yang berbeda, dapat mengurangi kinerja puncak yang dapat dicapai instance jika aliran dilacak. Untuk informasi selengkapnya tentang pelacakan koneksi grup keamanan, koneksi yang tidak dilacak, dan koneksi yang dilacak secara otomatis, lihatPelacakan koneksi grup keamanan Amazon EC2.

#### Driver jaringan

Driver jaringan diperbarui dan dirilis secara teratur. Jika driver Anda kedaluwarsa, itu dapat secara signifikan mengganggu kinerja. Perbarui driver Anda untuk memastikan bahwa Anda memiliki patch terbaru dan dapat memanfaatkan peningkatan kinerja, seperti fitur jalur akselerasi yang hanya tersedia untuk driver generasi terbaru. Driver sebelumnya tidak mendukung fitur jalur yang dipercepat.

Untuk memanfaatkan fitur jalur yang dipercepat, sebaiknya instal driver ENA terbaru pada instans Anda.

Instans Linux — driver ENA Linux 2.2.9 atau yang lebih baru. Untuk menginstal atau memperbarui driver ENA Linux dari GitHub repositori Amazon Drivers, lihat bagian kompilasi driver dari file readme.

Instans Windows - ENA driver Windows 2.0.0 atau yang lebih baru. Untuk menginstal atau memperbarui driver ENA Windows, lihatInstal driver ENA pada instans EC2 Windows.

#### Jarak antar titik akhir

Koneksi antara dua instance di Zona Ketersediaan yang sama dapat memproses lebih banyak paket per detik daripada koneksi lintas Wilayah sebagai hasil dari jendela TCP di lapisan aplikasi, yang menentukan berapa banyak data yang dapat terbang pada waktu tertentu. Jarak yang jauh antara instance meningkatkan latensi dan mengurangi jumlah paket yang dapat diproses oleh titik akhir.

### Batas antrian byte (BQL)

BQL adalah fitur yang membatasi jumlah byte yang diteruskan ke kartu Nitro untuk mengurangi antrian. BQL dinonaktifkan secara default di driver ENA, di sistem operasi Amazon Linux, dan di sebagian besar distribusi Linux. Jika BQL dan penggantian proxy fragmen keduanya diaktifkan, itu dapat mengakibatkan keterbatasan kinerja dengan membatasi jumlah byte yang diteruskan ke Nitro sebelum semua fragmen diproses.

## Pertimbangan desain aplikasi

Ada aspek desain aplikasi dan konfigurasi yang dapat mempengaruhi efisiensi pemrosesan Anda. Daftar berikut mencakup beberapa pertimbangan penting.

### Ukuran paket

Ukuran paket yang lebih besar dapat meningkatkan throughput untuk data yang dapat dikirim dan diterima instans di jaringan. Amazon EC2 mendukung bingkai jumbo sebesar 9001 byte, namun layanan lain mungkin menerapkan batasan yang berbeda. Ukuran paket yang lebih kecil dapat meningkatkan kecepatan proses paket, tetapi ini dapat mengurangi bandwidth maksimum yang dicapai ketika jumlah paket melebihi tunjangan PPS.

Jika ukuran paket melebihi Unit Transmisi Maksimum (MTU) dari hop jaringan, router di sepanjang jalur mungkin memecahnya. Fragmen paket yang dihasilkan dianggap pengecualian, dan biasanya diproses pada tingkat standar (tidak dipercepat). Hal ini dapat menyebabkan variasi dalam kinerja Anda. Namun, Anda dapat mengganti perilaku standar untuk paket terfragmentasi keluar dengan pengaturan mode proxy fragmen. Untuk informasi selengkapnya, lihat Maksimalkan kinerja jaringan pada sistem Nitro Anda. Kami menyarankan Anda mengevaluasi topologi Anda ketika Anda mengkonfigurasi MTU.

### Pengorbanan protokol

Protokol yang andal seperti TCP memiliki lebih banyak overhead daripada protokol yang tidak dapat diandalkan seperti UDP. Overhead yang lebih rendah dan pemrosesan jaringan yang disederhanakan untuk protokol transport UDP dapat menghasilkan tingkat PPS yang lebih tinggi, tetapi dengan mengorbankan pengiriman paket yang andal. Jika pengiriman paket yang andal tidak penting untuk aplikasi Anda, UDP mungkin merupakan pilihan yang baik.

### Micro-bursting

Micro-bursting terjadi ketika lalu lintas melebihi tunjangan selama periode waktu yang singkat daripada didistribusikan secara merata. Ini biasanya terjadi pada skala mikrodetik.

Misalnya, Anda memiliki instance yang dapat mengirim hingga 10 Gbps, dan aplikasi Anda mengirimkan 10 Gb penuh dalam setengah detik. Ledakan mikro ini melebihi tunjangan selama setengah detik pertama dan tidak menyisakan apa pun untuk sisa detik kedua. Meskipun Anda mengirim 10Gb dalam jangka waktu 1 detik, tunjangan dalam setengah detik pertama dapat mengakibatkan paket mengantri atau dijatuhkan.

Anda dapat menggunakan penjadwal jaringan seperti Linux Traffic Control untuk membantu mempercepat throughput Anda dan menghindari menyebabkan paket yang mengantri atau jatuh sebagai akibat dari micro-bursting.

Jumlah aliran

Satu aliran dibatasi hingga 5 Gbps kecuali berada di dalam grup penempatan cluster yang mendukung hingga 10 Gbps, atau jika menggunakan ENA Express, yang mendukung hingga 25 Gbps.

Demikian pula, kartu Nitro dapat memproses lebih banyak paket di beberapa aliran daripada menggunakan aliran tunggal. Untuk mencapai tingkat pemrosesan paket puncak per instans, kami merekomendasikan setidaknya 100 aliran pada instans dengan bandwidth agregat 100 Gbps atau lebih tinggi. Ketika kemampuan bandwidth agregat meningkat, jumlah aliran yang dibutuhkan untuk mencapai tingkat pemrosesan puncak juga meningkat. Benchmarking akan membantu Anda menentukan konfigurasi apa yang Anda butuhkan untuk mencapai tingkat puncak di jaringan Anda.

Antrian Adaptor Jaringan Elastis (ENA)

ENA (Elastic Network Adapter) menggunakan antrian multiple receiver (Rx) dan transmisi (Tx) (ENA antrean) untuk meningkatkan kinerja jaringan dan skalabilitas pada instans EC2. Antrian ini secara efisien mengelola lalu lintas jaringan dengan menyeimbangkan beban data yang dikirim dan diterima di seluruh antrian yang tersedia.

Untuk informasi selengkapnya, lihat the section called “Antrian ENA”.

Overhead proses fitur

Fitur seperti Traffic Mirroring dan ENA Express dapat menambahkan lebih banyak overhead pemrosesan, yang dapat mengurangi kinerja pemrosesan paket absolut. Anda dapat membatasi penggunaan fitur atau menonaktifkan fitur untuk meningkatkan kecepatan pemrosesan paket.

Pelacakan koneksi untuk mempertahankan status

Grup keamanan Anda menggunakan pelacakan koneksi untuk menyimpan informasi tentang lalu lintas ke dan dari instance. Pelacakan koneksi menerapkan aturan terhadap setiap aliran lalu

lintas jaringan individu untuk menentukan apakah lalu lintas diizinkan atau ditolak. Kartu Nitro menggunakan pelacakan aliran untuk mempertahankan keadaan aliran. Karena lebih banyak aturan grup keamanan diterapkan, lebih banyak pekerjaan diperlukan untuk mengevaluasi alur.

> **Note**
>
> Tidak semua arus lalu lintas jaringan dilacak. Jika aturan grup keamanan dikonfigurasi dengan Koneksi-koneksi yang tidak dilacak, tidak ada pekerjaan tambahan yang diperlukan kecuali untuk koneksi yang dilacak secara otomatis untuk memastikan perutean simetris ketika ada beberapa jalur balasan yang valid.

Paket yang tidak menggunakan akselerasi perangkat keras

Tidak semua paket dapat memanfaatkan akselerasi perangkat keras. Penanganan pengecualian ini melibatkan beberapa overhead pemrosesan yang diperlukan untuk memastikan kesehatan aliran jaringan Anda. Aliran jaringan harus dapat diandalkan memenuhi standar protokol, sesuai dengan perubahan dalam desain VPC, dan merutekan paket hanya ke tujuan yang diizinkan. Namun, overhead mengurangi kinerja Anda.

Fragmen paket

Seperti disebutkan di bawah  Pertimbangan Aplikasi, fragmen paket yang dihasilkan dari paket yang melebihi MTU jaringan biasanya ditangani sebagai pengecualian, dan tidak dapat memanfaatkan akselerasi perangkat keras. Namun, Anda dapat melewati batasan fragmen keluar dengan mode proxy fragmen, tergantung pada versi driver Anda. Untuk informasi selengkapnya, lihat tindakan yang dapat Anda lakukan di Maksimalkan kinerja jaringan pada sistem Nitro Anda bagian ini.

Koneksi idle

Ketika koneksi tidak memiliki aktivitas untuk sementara waktu, bahkan jika koneksi belum mencapai batas waktu tunggu, sistem dapat menurunkan prioritasnya. Kemudian, jika data masuk setelah koneksi tidak diprioritaskan, sistem perlu menanganinya sebagai pengecualian untuk menyambung kembali.

Untuk mengelola koneksi, Anda dapat menggunakan batas waktu pelacakan koneksi untuk menutup koneksi idle. Anda juga dapat menggunakan keepalives TCP untuk menjaga koneksi idle tetap terbuka. Untuk informasi selengkapnya, lihat Waktu habis pelacakan koneksi idle.

Mutasi VPC

Pembaruan untuk grup keamanan, tabel rute, dan daftar kontrol akses semuanya perlu dievaluasi ulang di jalur pemrosesan untuk memastikan bahwa entri rute dan aturan grup keamanan masih berlaku seperti yang diharapkan.

Aliran ICMP

Internet Control Message Protocol (ICMP) adalah protokol lapisan jaringan yang digunakan perangkat jaringan untuk mendiagnosis masalah komunikasi jaringan. Paket-paket ini selalu menggunakan aliran penuh.

Aliran L2 asimetris

NitroV3 dan platform sebelumnya tidak menggunakan akselerasi perangkat keras untuk lalu lintas antara dua ENI di subnet yang sama di mana satu ENI menggunakan router gateway default dan yang lainnya tidak. NitroV4 dan platform yang lebih baru menggunakan akselerasi perangkat keras dalam skenario ini. Untuk kinerja yang lebih baik pada NitroV3 atau platform sebelumnya, pastikan bahwa router gateway default menggunakan kecocokan antara kedua ENI, atau ENI tersebut berada di subnet yang berbeda.

## Maksimalkan kinerja jaringan pada sistem Nitro Anda

Anda dapat memaksimalkan kinerja jaringan Anda pada sistem Nitro dengan menyesuaikan pengaturan jaringan.

Topik

- Pertimbangan-pertimbangan
- Menyetel kinerja PPS
- Konfigurasikan alokasi antrian ENA
- Memantau kinerja pada instans Linux

### Pertimbangan-pertimbangan

Sebelum Anda membuat keputusan desain atau menyesuaikan pengaturan jaringan apa pun pada instans Anda, kami sarankan Anda mengambil langkah-langkah berikut untuk membantu memastikan bahwa Anda mendapatkan hasil terbaik:

1. Pahami pro dan kontra dari tindakan yang dapat Anda ambil untuk meningkatkan kinerja dengan meninjauPertimbangan desain jaringan untuk sistem Nitro.

   Untuk pertimbangan lebih lanjut dan praktik terbaik untuk konfigurasi instans Anda di Linux, lihat Praktik Terbaik Driver Linux ENA dan Panduan Optimasi Kinerja di GitHub.

2. Tolok ukur beban kerja Anda dengan jumlah aliran aktif puncak untuk menentukan baseline kinerja aplikasi Anda. Dengan dasar kinerja, Anda dapat menguji variasi dalam pengaturan atau desain aplikasi untuk memahami pertimbangan mana yang paling berdampak, terutama jika Anda berencana untuk meningkatkan atau memperkecil skala.

### Menyetel kinerja PPS

Daftar berikut berisi tindakan yang dapat Anda lakukan untuk menyesuaikan kinerja PPS Anda, tergantung pada kebutuhan sistem Anda.

- Kurangi jarak fisik antara dua contoh. Saat instans pengiriman dan penerimaan berada di Availability Zone yang sama atau menggunakan grup penempatan cluster, Anda dapat mengurangi jumlah lompatan yang diperlukan paket untuk melakukan perjalanan dari satu titik akhir ke titik akhir lainnya.
- Gunakan Koneksi-koneksi yang tidak dilacak.
- Gunakan protokol UDP untuk lalu lintas jaringan.
- Untuk instans EC2 dengan bandwidth agregat 100 Gbps atau lebih, distribusikan beban kerja lebih dari 100 aliran individu atau lebih untuk menyebarkan pekerjaan secara merata di seluruh kartu Nitro.
- Untuk mengatasi batas PPS fragmen keluar pada instans EC2, Anda dapat mengaktifkan mode proxy fragmen (tergantung pada versi driver Anda). Pengaturan ini memungkinkan paket terfragmentasi dievaluasi di jalur pemrosesan, sehingga mengatasi batas PPS keluar 1024. Saat memuat driver, jalankan salah satu perintah berikut untuk mengaktifkan atau menonaktifkan mode proxy fragmen:

  Aktifkan mode proxy fragmen

  ```
  sudo insmod ena.ko enable_frag_bypass=1
  ```

  Nonaktifkan mode proxy fragmen

```
sudo insmod ena.ko enable_frag_bypass=0
```

Konfigurasikan alokasi antrian ENA

Pada jenis instans yang didukung, Anda dapat mengalokasikan antrian ini secara dinamis di seluruh Elastic Network Interfaces (ENI). Alokasi antrian ENA yang fleksibel mengoptimalkan distribusi sumber daya, memungkinkan pemanfaatan vCPU maksimum. Beban kerja kinerja jaringan tinggi biasanya memerlukan beberapa antrian ENA. Untuk informasi selengkapnya, lihat Antrian ENA.

Memantau kinerja pada instans Linux

Anda dapat menggunakan metrik Ethtool pada instans Linux untuk memantau indikator kinerja jaringan instance seperti bandwidth, kecepatan paket, dan pelacakan koneksi. Lihat informasi yang lebih lengkap di Memantau kinerja jaringan untuk pengaturan ENA pada instans EC2 Anda.

## Mengoptimalkan kinerja jaringan pada instans EC2 Windows

Untuk mencapai kinerja jaringan maksimum pada instans Windows Anda dengan jaringan yang disempurnakan, Anda mungkin perlu memodifikasi konfigurasi sistem operasi default. Kami merekomendasikan perubahan konfigurasi berikut untuk aplikasi yang memerlukan performa jaringan tinggi. Pengoptimalan lain (seperti mengaktifkan checksum offloading dan mengaktifkan RSS, misalnya) sudah dikonfigurasi pada AMI Windows resmi.

> **Note**
>
> TCP chimney offloading harus dinonaktifkan di sebagian besar kasus penggunaan, dan sudah tidak digunakan lagi mulai Windows Server 2016.

Selain pengoptimalan sistem operasi ini, Anda juga harus mempertimbangkan unit transmisi maksimum (MTU) lalu lintas jaringan Anda, dan menyesuaikannya dengan beban kerja dan arsitektur jaringan Anda. Untuk informasi selengkapnya, lihat Maximum transmission unit (MTU) jaringan untuk instans EC2 Anda.

AWS secara teratur mengukur latensi perjalanan pulang pergi rata-rata antara instance yang diluncurkan dalam grup penempatan cluster 50us dan latensi ekor 200us pada persentil 99,9. Jika aplikasi Anda membutuhkan latensi rendah secara konsisten, kami merekomendasikan menggunakan driver ENA versi terbaru pada instans berbasis Nitro dengan performa tetap.

## Konfigurasikan afinitas CPU penskalaan sisi Terima

Receive side scaling (RSS) digunakan untuk mendistribusikan beban CPU lalu lintas jaringan ke beberapa prosesor. Secara default, Amazon Windows AMI yang resmi dikonfigurasi dengan mengaktifkan RSS. Antarmuka jaringan elastis ENA menyediakan hingga delapan antrian RSS. Dengan menentukan afinitas CPU untuk antrean RSS, serta untuk proses sistem lainnya, dimungkinkan untuk menyebarkan beban CPU melalui sistem multi-core, yang memungkinkan lebih banyak lalu lintas jaringan untuk diproses. Pada tipe instance dengan lebih dari 16 vCPU, sebaiknya gunakan `Set-NetAdapterRSS` PowerShell cmdlet, yang secara manual mengecualikan prosesor boot (prosesor logis 0 dan 1 saat hyper-threading diaktifkan) dari konfigurasi RSS untuk semua antarmuka jaringan elastis, untuk mencegah perselisihan dengan berbagai komponen sistem.

Windows sadar hyper-thread dan memastikan bahwa antrian RSS dari kartu antarmuka jaringan tunggal (NIC) selalu ditempatkan pada inti fisik yang berbeda. Oleh karena itu, kecuali hyper-threading dinonaktifkan, untuk sepenuhnya mencegah perselisihan dengan NIC lain, sebarkan konfigurasi RSS dari setiap NIC di antara rentang 16 prosesor logis. `Set-NetAdapterRss`Cmdlet memungkinkan Anda untuk menentukan rentang per-NIC dari prosesor logis yang valid dengan menentukan nilai BaseProcessorGroup,,, BaseProcessorNumber MaxProcessingGroup MaxProcessorNumber, dan NumaNode (opsional). Jika tidak ada cukup inti fisik untuk sepenuhnya menghilangkan perselisihan antar-NIC, minimalkan rentang yang tumpang tindih atau kurangi jumlah prosesor logis dalam rentang antarmuka jaringan elastis tergantung pada beban kerja antarmuka yang diharapkan (dengan kata lain, antarmuka jaringan administratif volume rendah mungkin tidak memerlukan banyak antrian RSS yang ditetapkan). Selain itu, seperti disebutkan sebelumnya, berbagai komponen harus dijalankan pada CPU 0, oleh karena itu kami menyarankan untuk mengecualikannya dari semua konfigurasi RSS jika vCPU cukup tersedia.

Misalnya, ketika ada tiga antarmuka jaringan elastis pada 72 instance vCPU dengan 2 node NUMA dengan hyper-threading diaktifkan, perintah berikut menyebarkan beban jaringan antara dua CPU tanpa tumpang tindih dan mencegah penggunaan inti 0 sepenuhnya.

```
Set-NetAdapterRss -Name NIC1 -BaseProcessorGroup 0 -BaseProcessorNumber 2 -
MaxProcessorNumber 16
Set-NetAdapterRss -Name NIC2 -BaseProcessorGroup 1 -BaseProcessorNumber 0 -
MaxProcessorNumber 14
Set-NetAdapterRss -Name NIC3 -BaseProcessorGroup 1 -BaseProcessorNumber 16 -
MaxProcessorNumber 30
```

Perhatikan bahwa pengaturan ini tetap ada untuk setiap adaptor jaringan. Jika ukuran instance diubah menjadi satu dengan jumlah vCPU yang berbeda, Anda harus mengevaluasi kembali

konfigurasi RSS untuk setiap antarmuka jaringan elastis yang diaktifkan. Dokumentasi Microsoft lengkap untuk cmdlet dapat ditemukan di sini:. Set-NetAdapterRss

Catatan khusus untuk beban kerja SQL: Kami juga menyarankan Anda meninjau pengaturan afinitas utas I/O Anda bersama dengan konfigurasi RSS antarmuka jaringan elastis Anda untuk meminimalkan I/O dan perselisihan jaringan untuk CPU yang sama. Lihat Konfigur asi server: topeng afinitas.

# Adaptor Kain Elastis untuk AI/ML dan beban kerja HPC di Amazon EC2

Elastic Fabric Adapter (EFA) adalah perangkat jaringan yang dapat Anda lampirkan ke instans Amazon EC2 untuk mempercepat aplikasi Artificial Intelligence (AI), Machine Learning (ML), dan High Performance Computing (HPC). EFA memungkinkan Anda mencapai kinerja aplikasi cluster lokal AI/ML atau HPC, dengan skalabilitas, fleksibilitas, dan elastisitas yang disediakan oleh Cloud. AWS

EFA memberikan latensi yang lebih rendah dan lebih konsisten serta throughput yang lebih tinggi dibandingkan transportasi TCP yang secara tradisional digunakan dalam sistem HPC berbasis cloud. Ini meningkatkan kinerja komunikasi antar instans yang sangat penting untuk penskalaan AI/ML dan aplikasi HPC. Ini dioptimalkan untuk bekerja pada infrastruktur AWS jaringan yang ada dan dapat ditingkatkan tergantung pada persyaratan aplikasi.

EFA terintegrasi dengan Libfabric, dan mendukung Nvidia Collective Communications Library (NCCL) dan NVIDIA Inference Xfer Library (NIXL) untuk aplikasi AI dan ML, dan Open MPI 4.1 dan yang lebih baru dan Intel MPI 2019 Update 5 dan yang lebih baru untuk aplikasi HPC. NCCL dan MPI terintegrasi dengan Libfabric 1.7.0 dan yang lebih baru. NIXL terintegrasi dengan Libfabric 1.21.0 dan yang lebih baru.

EFA mendukung penulisan RDMA (Remote Direct Memory Access) pada sebagian besar jenis instans yang didukung yang memiliki Nitro versi 4 dan yang lebih baru. Pembacaan RDMA didukung pada semua instance dengan Nitro versi 4 dan yang lebih baru. Untuk informasi selengkapnya, lihat Tipe instans yang didukung.

Daftar Isi


- Dasar-dasar EFA
- Antarmuka dan pustaka yang didukung
- Tipe instans yang didukung

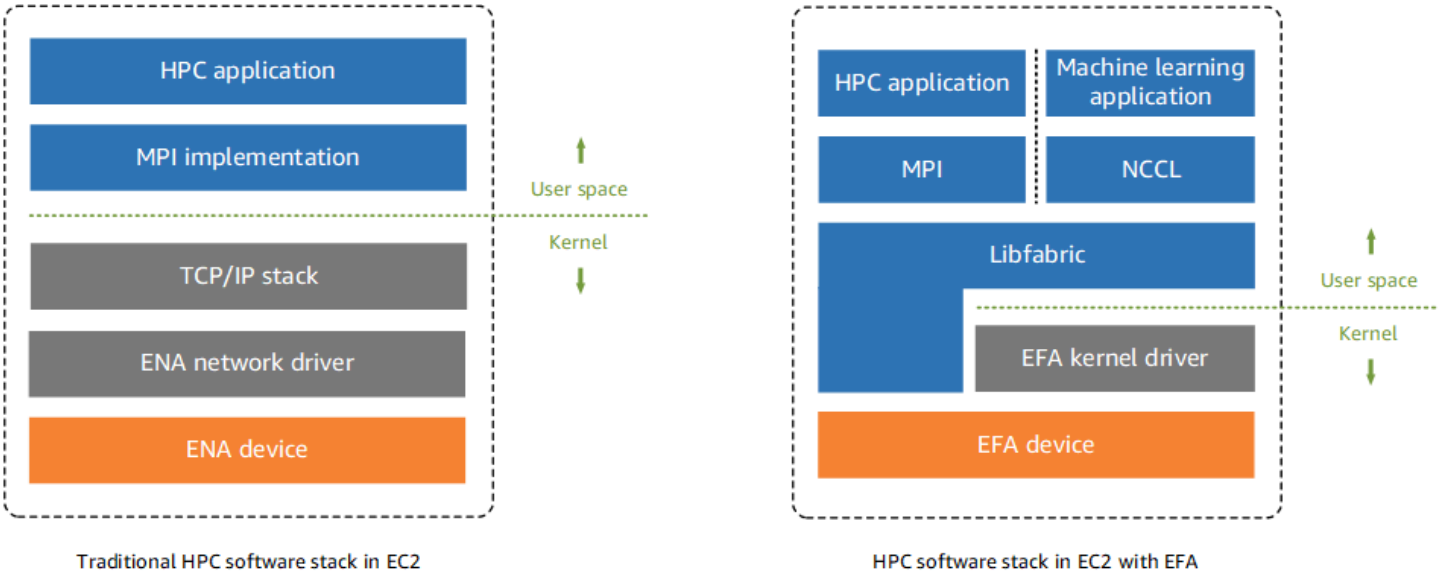
- Sistem operasi yang didukung
- Batasan EFA
- Harga EFA
- Memulai dengan EFA dan MPI untuk beban kerja HPC di Amazon EC2
- Memulai dengan EFA dan NCCL untuk beban kerja ML di Amazon EC2
- Memulai dengan EFA dan NIXL untuk beban kerja inferensi di Amazon EC2
- Maksimalkan bandwidth jaringan pada instans Amazon EC2 dengan beberapa kartu jaringan
- Instal perangkat lunak EFA dalam wadah
- Membuat dan memasang Adaptor Kain Elastis ke instans Amazon EC2
- Melepaskan dan menghapus EFA dari instans Amazon EC2
- Pantau Adaptor Kain Elastis di Amazon EC2
- Memverifikasi penginstal EFA menggunakan checksum
- Catatan rilis Adaptor Kain Elastis

## Dasar-dasar EFA

Perangkat EFA dapat dilampirkan ke instance EC2 dengan dua cara:

1. Menggunakan antarmuka EFA tradisional, juga disebut EFA dengan ENA, yang menciptakan perangkat EFA dan perangkat ENA.
2. Menggunakan EFA-only antarmuka, yang hanya menciptakan perangkat EFA.

Perangkat EFA menyediakan kemampuan seperti kontrol bawaan OS-bypass dan kemacetan melalui protokol Scalable Reliable Datagram (SRD). Fitur perangkat EFA memungkinkan fungsi transportasi yang andal dan latensi rendah yang memungkinkan antarmuka EFA memberikan kinerja aplikasi yang lebih baik untuk aplikasi HPC dan ML di Amazon EC2. Sementara perangkat ENA menawarkan jaringan IP tradisional. Untuk panduan tentang konfigurasi instans mana yang akan digunakan berdasarkan kasus penggunaan Anda, lihatMaksimalkan bandwidth jaringan.




Secara tradisional, AI/ML aplikasi menggunakan NCCL dan NIXL (untuk inferensi terpilah). Aplikasi HPC menggunakan Message Passing Interface (MPI) untuk berinteraksi dengan transportasi jaringan sistem. Di AWS cloud, ini berarti bahwa aplikasi berinteraksi dengan NCCL, NIXL, atau MPI, yang kemudian menggunakan tumpukan sistem operasi dan TCP/IP driver perangkat ENA untuk mengaktifkan komunikasi jaringan antar instance.

Dengan EFA tradisional (EFA dengan ENA) atau EFA-only antarmuka, AI/ML aplikasi menggunakan NCCL dan NIXL (untuk inferensi terpilah). Aplikasi HPC menggunakan MPI, untuk berinteraksi langsung dengan Libfabric API. API Libfabric memintas kernel sistem operasi dan berkomunikasi secara langsung dengan perangkat ITEFA untuk menempatkan paket pada jaringan. Ini mengurangi overhead dan memungkinkan AI/ML dan aplikasi HPC berjalan lebih efisien.

> **Note**
>
> Libfabric adalah komponen inti dari kerangka Inter OpenFabrics faces (OFI), yang mendefinisikan dan mengekspor API ruang pengguna OFI. Untuk informasi lebih lanjut, lihat situs web Libfabric OpenFabrics.

## Perbedaan antara ENA, EFA, dan antarmuka EFA-only jaringan

Amazon EC2 menyediakan dua jenis antarmuka jaringan:

- Antar  muka ENA menyediakan semua jaringan IP tradisional dan fitur routing yang diperlukan untuk mendukung jaringan IP untuk VPC. Untuk informasi selengkapnya, lihat Aktifkan jaringan yang disempurnakan dengan ENA pada instans EC2 Anda.
- Antarmuka EFA  (EFA dengan ENA) menyediakan perangkat ENA untuk jaringan IP dan perangkat EFA untuk komunikasi throughput tinggi dengan latensi rendah.
- EFA-onlyantarmuka hanya mendukung kemampuan perangkat EFA, tanpa perangkat ENA untuk jaringan IP tradisional.

Tabel berikut memberikan perbandingan ENA, EFA, dan antarmuka EFA-only jaringan.

| | ENA | EFA (EFA dengan ENA) | EFA-only |
|---|---|---|---|
| Mendukung fungsionalitas jaringan IP | Ya | Ya | Tidak |
| Dapat ditetapkan alamat IPv4 atau IPv6 | Ya | Ya | Tidak |
| Dapat digunakan sebagai antarmuka jaringan utama misalnya | Ya | Ya | Tidak |
| Menghitung batas lampiran ENI misalnya | Ya | Ya | Ya |
| Dukungan jenis instans | Didukung pada semua jenis Nitro-based instans | Jenis instans yang didukung | Jenis instans yang didukung |

| | ENA | EFA (EFA dengan ENA) | EFA-only |
|---|---|---|---|
| Penamaan parameter di API EC2 | `interface` | `efa` | `efa-only` |
| Penamaan bidang di konsol EC2 | Tidak ada pilihan | EFA dengan ENA | EFA-only |

## Antarmuka dan pustaka yang didukung

EFA mendukung antarmuka dan pustaka berikut:

- Buka MPI 4.1 dan yang lebih baru
- Pembaruan 5 Intel MPI 2019 dan versi yang lebih baru
- NVIDIA Collective Communications Library (NCCL) 2.4.2 dan yang lebih baru
- NVIDIA Inference Xfer Library (NIXL) 1.0.0 dan yang lebih baru
- AWS Neuron SDK versi 2.3 dan yang lebih baru

## Tipe instans yang didukung

Semua jenis instance berikut mendukung EFA. Selain itu, tabel menunjukkan dukungan baca RDMA dan penulisan RDMA untuk jenis instance.

Nitro v6 (EFA v4)

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| Tujuan Umum | | |
| m8a.48xbesar | Ya | Ya |
| m8a.logam-48xl | Ya | Ya |
| m8azn.24xbesar | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| m8azn.logam-24xl | Ya | Ya |
| m8gb.16xbesar | Ya | Ya |
| m8gb.24xbesar | Ya | Ya |
| m8gb.48xbesar | Ya | Ya |
| m8gb.logam-24xl | Ya | Ya |
| m8gb.logam-48xl | Ya | Ya |
| m8gn.16xbesar | Ya | Ya |
| m8gn.24xbesar | Ya | Ya |
| m8gn.48xbesar | Ya | Ya |
| m8gn.logam-24xl | Ya | Ya |
| m8gn.logam-48xl | Ya | Ya |
| m8i.48xbesar | Ya | Ya |
| m8i.96xbesar | Ya | Ya |
| m8i.logam-48xl | Ya | Ya |
| m8i.logam-96xl | Ya | Ya |
| m8id.48xbesar | Ya | Ya |
| m8id.96xbesar | Ya | Ya |
| m8id.logam-48xl | Ya | Ya |
| m8id.logam-96xl | Ya | Ya |
| m8in.48xbesar | Ya | Ya |
| m8in.96xbesar | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| m8in.logam-48xl | Ya | Ya |
| m8in.logam-96xl | Ya | Ya |
| m8idn.48xbesar | Ya | Ya |
| m8idn.96xbesar | Ya | Ya |
| m8idn.logam-48xl | Ya | Ya |
| m8idn.logam-96xl | Ya | Ya |
| m8ib.48xbesar | Ya | Ya |
| m8ib.96xbesar | Ya | Ya |
| m8ib.logam-48xl | Ya | Ya |
| m8ib.logam-96xl | Ya | Ya |
| m8idb.48xbesar | Ya | Ya |
| m8idb.96xbesar | Ya | Ya |
| m8idb.logam-48xl | Ya | Ya |
| m8idb.logam-96xl | Ya | Ya |
| m9g.48xbesar | Ya | Ya |
| m9g.logam-48xl | Ya | Ya |
| m9gd.48xbesar | Ya | Ya |
| m9gd.logam-48xl | Ya | Ya |
| Komputasi yang Dioptimalkan | | |
| c8a.48xbesar | Ya | Ya |
| c8a.logam-48xl | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| c8gb.16xbesar | Ya | Ya |
| c8gb.24xbesar | Ya | Ya |
| c8gb.48xbesar | Ya | Ya |
| c8gb.logam-24xl | Ya | Ya |
| c8gb.logam-48xl | Ya | Ya |
| c8gn.16xbesar | Ya | Ya |
| c8gn.24xbesar | Ya | Ya |
| c8gn.48xbesar | Ya | Ya |
| c8gn.logam-24xl | Ya | Ya |
| c8gn.logam-48xl | Ya | Ya |
| c8i.48xbesar | Ya | Ya |
| c8i.96xbesar | Ya | Ya |
| c8i.logam-48xl | Ya | Ya |
| c8i.logam-96xl | Ya | Ya |
| c8id.48xbesar | Ya | Ya |
| c8id.96xbesar | Ya | Ya |
| c8id.logam-48xl | Ya | Ya |
| c8id.logam-96xl | Ya | Ya |
| c8in.48xbesar | Ya | Ya |
| c8in.96xbesar | Ya | Ya |
| c8in.logam-48xl | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| c8in.logam-96xl | Ya | Ya |
| c8ib.48xbesar | Ya | Ya |
| c8ib.96xbesar | Ya | Ya |
| c8ib.logam-48xl | Ya | Ya |
| c8ib.logam-96xl | Ya | Ya |
| c9g.48xbesar | Ya | Ya |
| c9g.logam-48xl | Ya | Ya |
| c9gd.48xbesar | Ya | Ya |
| c9gd.logam-48xl | Ya | Ya |
| Memori Dioptimalkan | | |
| r8a.48xbesar | Ya | Ya |
| r8a.logam-48xl | Ya | Ya |
| r8gb.16xbesar | Ya | Ya |
| r8gb.24xbesar | Ya | Ya |
| r8gb.48xbesar | Ya | Ya |
| r8gb.logam-24xl | Ya | Ya |
| r8gb.logam-48xl | Ya | Ya |
| r8gn.16xbesar | Ya | Ya |
| r8gn.24xbesar | Ya | Ya |
| r8gn.48xbesar | Ya | Ya |
| r8gn.logam-24xl | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| r8gn.logam-48xl | Ya | Ya |
| r8i.48xbesar | Ya | Ya |
| r8i.96xbesar | Ya | Ya |
| r8i.logam-48xl | Ya | Ya |
| r8i.logam-96xl | Ya | Ya |
| r8id.48xbesar | Ya | Ya |
| r8id.96xbesar | Ya | Ya |
| r8id.logam-48xl | Ya | Ya |
| r8id.logam-96xl | Ya | Ya |
| r8in.48xbesar | Ya | Ya |
| r8in.96xbesar | Ya | Ya |
| r8in.logam-48xl | Ya | Ya |
| r8in.logam-96xl | Ya | Ya |
| r8idn.48xbesar | Ya | Ya |
| r8idn.96xbesar | Ya | Ya |
| r8idn.logam-48xl | Ya | Ya |
| r8idn.logam-96xl | Ya | Ya |
| r8ib.48xbesar | Ya | Ya |
| r8ib.96xbesar | Ya | Ya |
| r8ib.logam-48xl | Ya | Ya |
| r8ib.logam-96xl | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| r8idb.48xbesar | Ya | Ya |
| r8idb.96xbesar | Ya | Ya |
| r8idb.logam-48xl | Ya | Ya |
| r8idb.logam-96xl | Ya | Ya |
| r9g.48xbesar | Ya | Ya |
| r9g.logam-48xl | Ya | Ya |
| r9gd.48xbesar | Ya | Ya |
| r9gd.logam-48xl | Ya | Ya |
| x8aedz.24xbesar | Ya | Ya |
| x8aedz.logam-24xl | Ya | Ya |
| x8i.48xbesar | Ya | Ya |
| x8i.64xbesar | Ya | Ya |
| x8i.96xbesar | Ya | Ya |
| x8i.logam-48xl | Ya | Ya |
| x8i.logam-96xl | Ya | Ya |
| Penyimpanan Dioptimalkan | | |
| i8ge.48xbesar | Ya | Ya |
| i8ge.logam-48xl | Ya | Ya |
| Komputasi yang Dipercepat | | |
| g7.8xbesar | Ya | Ya |
| g7.12xbesar | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| g7.24xbesar | Ya | Ya |
| g7.48xbesar | Ya | Ya |
| g7e.8xbesar | Ya | Ya |
| g7e.12xbesar | Ya | Ya |
| g7e.24xbesar | Ya | Ya |
| g7e.48xbesar | Ya | Ya |
| p6-b200.48xbesar | Ya | Ya |
| p6-b300.48xbesar | Ya | Ya |
| Komputasi Performa Tinggi | | |
| hpc8a.96xbesar | Ya | Ya |

Nitro v5 (EFA v3)

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| Tujuan Umum | | |
| m8g.24xbesar | Ya | Ya |
| m8g.48xbesar | Ya | Ya |
| m8g.logam-24xl | Ya | Ya |
| m8g.logam-48xl | Ya | Ya |
| m8gd.24xbesar | Ya | Ya |
| m8gd.48xbesar | Ya | Ya |
| m8gd.logam-24xl | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| m8gd.logam-48xl | Ya | Ya |
| Komputasi yang Dioptimalkan | | |
| c7gn.16xlarge | Ya | Tidak |
| c7gn.logam | Ya | Tidak |
| c8g.24xbesar | Ya | Ya |
| c8g.48xbesar | Ya | Ya |
| c8g.logam-24xl | Ya | Ya |
| c8g.logam-48xl | Ya | Ya |
| c8gd.24xbesar | Ya | Ya |
| c8gd.48xbesar | Ya | Ya |
| c8gd.logam-24xl | Ya | Ya |
| c8gd.logam-48xl | Ya | Ya |
| Memori Dioptimalkan | | |
| r8g.24xbesar | Ya | Ya |
| r8g.48xbesar | Ya | Ya |
| r8g.logam-24xl | Ya | Ya |
| r8g.logam-48xl | Ya | Ya |
| r8gd.24xbesar | Ya | Ya |
| r8gd.48xbesar | Ya | Ya |
| r8gd.logam-24xl | Ya | Ya |
| r8gd.logam-48xl | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| x8g.24xbesar | Ya | Ya |
| x8g.48xbesar | Ya | Ya |
| x8g.logam-24xl | Ya | Ya |
| x8g.logam-48xl | Ya | Ya |
| Penyimpanan Dioptimalkan | | |
| i7ie.48xbesar | Ya | Ya |
| i7ie.logam-48xl | Ya | Ya |
| i8g.48xbesar | Ya | Ya |
| i8g.logam-48xl | Ya | Ya |
| Komputasi yang Dipercepat | | |
| p5en.48xbesar | Ya | Ya |
| p6e-gb200.36xbesar | Ya | Ya |
| trn2.3xbesar | Ya | Ya |
| trn2.48xbesar | Ya | Ya |
| trn2u.48xbesar | Ya | Ya |
| Komputasi Performa Tinggi | | |
| hpc7g.4xlarge | Ya | Tidak |
| hpc7g.8xlarge | Ya | Tidak |
| hpc7g.16xlarge | Ya | Tidak |

Nitro v4 (EFA v2)

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| Tujuan Umum | | |
| m6a.48xlarge | Ya | Ya |
| m6a.metal | Ya | Ya |
| m6i.32xlarge | Ya | Ya |
| m6i.metal | Ya | Ya |
| m6id.32xlarge | Ya | Ya |
| m6id.metal | Ya | Ya |
| m6idn.32xlarge | Ya | Ya |
| m6idn.metal | Ya | Ya |
| m6in.32xlarge | Ya | Ya |
| m6in.metal | Ya | Ya |
| m7a.48xlarge | Ya | Ya |
| m7a.metal-48xl | Ya | Ya |
| m7g.16xlarge | Ya | Ya |
| m7g.metal | Ya | Ya |
| m7gd.16xlarge | Ya | Ya |
| m7gd.logam | Ya | Ya |
| m7i.48xlarge | Ya | Ya |
| m7i.metal-48xl | Ya | Ya |
| Komputasi yang Dioptimalkan | | |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| c6a.48xlarge | Ya | Ya |
| c6a.metal | Ya | Ya |
| c6gn.16xlarge | Ya | Ya |
| c6i.32xlarge | Ya | Ya |
| c6i.metal | Ya | Ya |
| c6id.32xlarge | Ya | Ya |
| c6id.metal | Ya | Ya |
| c6in.32xlarge | Ya | Ya |
| c6in.metal | Ya | Ya |
| c7a.48xlarge | Ya | Ya |
| c7a.metal-48xl | Ya | Ya |
| c7g.16xlarge | Ya | Ya |
| c7g.metal | Ya | Ya |
| c7gd.16xlarge | Ya | Ya |
| c7gd.logam | Ya | Ya |
| c7i.48xlarge | Ya | Ya |
| c7i.metal-48xl | Ya | Ya |
| Memori Dioptimalkan | | |
| r6a.48xlarge | Ya | Ya |
| r6a.metal | Ya | Ya |
| r6i.32xlarge | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| r6i.metal | Ya | Ya |
| r6id.32xlarge | Ya | Ya |
| r6id.metal | Ya | Ya |
| r6idn.32xlarge | Ya | Ya |
| r6idn.metal | Ya | Ya |
| r6in.32xlarge | Ya | Ya |
| r6in.metal | Ya | Ya |
| r7a.48xlarge | Ya | Ya |
| r7a.metal-48xl | Ya | Ya |
| r7g.16xlarge | Ya | Ya |
| r7g.metal | Ya | Ya |
| r7gd.16xlarge | Ya | Ya |
| r7gd.metal | Ya | Ya |
| r7i.48xlarge | Ya | Ya |
| r7i.metal-48xl | Ya | Ya |
| r7iz.32xlarge | Ya | Ya |
| r7iz.metal-32xl | Ya | Ya |
| u7i-6tb.112xbesar | Ya | Ya |
| u7i-8tb.112xbesar | Ya | Ya |
| u7i-12tb.224xbesar | Ya | Ya |
| u7in-16tb.224xbesar | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| u7in-24tb.224xbesar | Ya | Ya |
| u7in-32tb.224xbesar | Ya | Ya |
| u7inh-32tb.480xbesar | Ya | Ya |
| x2idn.32xlarge | Ya | Ya |
| x2idn.metal | Ya | Ya |
| x2iedn.32xlarge | Ya | Ya |
| x2iedn.metal | Ya | Ya |
| Penyimpanan Dioptimalkan | | |
| i4g.16xlarge | Ya | Ya |
| i4i.32xlarge | Ya | Ya |
| i4i.metal | Ya | Ya |
| i7i.24xbesar | Ya | Ya |
| i7i.48xbesar | Ya | Ya |
| i7i.logam-48xl | Ya | Ya |
| im4gn.16xlarge | Ya | Ya |
| Komputasi yang Dipercepat | | |
| f2.48xbesar | Ya | Ya |
| g6.8xbesar | Ya | Ya |
| g6.12xbesar | Ya | Ya |
| g6.16xbesar | Ya | Ya |
| g6.24xbesar | Ya | Ya |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| g6.48xbesar | Ya | Ya |
| g6e.8xbesar | Ya | Ya |
| g6e.12xbesar | Ya | Ya |
| g6e.16xbesar | Ya | Ya |
| g6e.24xbesar | Ya | Ya |
| g6e.48xbesar | Ya | Ya |
| gr6.8xbesar | Ya | Ya |
| p5.4xbesar | Ya | Ya |
| p5.48xlarge | Ya | Ya |
| p5e.48xlarge | Ya | Ya |
| trn1.32xlarge | Ya | Ya |
| trn1n.32xlarge | Ya | Ya |
| Komputasi Performa Tinggi | | |
| hpc6a.48xlarge | Ya | Ya |
| hpc6id.32xlarge | Ya | Ya |
| hpc7a.12xlarge | Ya | Ya |
| hpc7a.24xlarge | Ya | Ya |
| hpc7a.48xlarge | Ya | Ya |
| hpc7a.96xlarge | Ya | Ya |

Nitro v3 (EFA v1)

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
|---|---|---|
| Tujuan Umum | | |
| m5dn.24xlarge | Tidak | Tidak |
| m5dn.metal | Tidak | Tidak |
| m5n.24xlarge | Tidak | Tidak |
| m5n.metal | Tidak | Tidak |
| m5zn.12xlarge | Tidak | Tidak |
| m5zn.metal | Tidak | Tidak |
| Komputasi yang Dioptimalkan | | |
| c5n.9xlarge | Tidak | Tidak |
| c5n.18xlarge | Tidak | Tidak |
| c5n.metal | Tidak | Tidak |
| Memori Dioptimalkan | | |
| r5dn.24xlarge | Tidak | Tidak |
| r5dn.metal | Tidak | Tidak |
| r5n.24xlarge | Tidak | Tidak |
| r5n.metal | Tidak | Tidak |
| x2iezn.12xlarge | Tidak | Tidak |
| x2iezn.metal | Tidak | Tidak |
| Penyimpanan Dioptimalkan | | |
| i3en.12xlarge | Tidak | Tidak |

| Tipe instans | Dukungan baca RDMA | Dukungan penulisan RDMA |
| --- | --- | --- |
| i3en.24xlarge | Tidak | Tidak |
| i3en.metal | Tidak | Tidak |
| Komputasi yang Dipercepat | | |
| dl2q.24xlarge | Tidak | Tidak |
| g4dn.8xlarge | Tidak | Tidak |
| g4dn.12xlarge | Tidak | Tidak |
| g4dn.16xlarge | Tidak | Tidak |
| g4dn.metal | Tidak | Tidak |
| g5.8xlarge | Tidak | Tidak |
| g5.12xlarge | Tidak | Tidak |
| g5.16xlarge | Tidak | Tidak |
| g5.24xlarge | Tidak | Tidak |
| g5.48xlarge | Tidak | Tidak |
| inf1.24xlarge | Tidak | Tidak |
| p3dn.24xlarge | Tidak | Tidak |
| p4d.24xlarge | Ya | Tidak |
| p4de.24xlarge | Ya | Tidak |
| vt1.24xlarge | Tidak | Tidak |
| Generasi Sebelumnya | | |
| p3dn.24xlarge | Tidak | Tidak |

Untuk melihat tipe instans yang tersedia yang mendukung EFA di Wilayah tertentu

Tipe instans yang tersedia berbeda-beda menurut Wilayah. Untuk melihat tipe instans yang tersedia yang mendukung EFA di suatu Wilayah, gunakan perintah describe-instance-types dengan parameter `--region`. Sertakan parameter `--filters` untuk cakupan hasil ke tipe instans yang mendukung EFA dan `--query` parameter untuk cakupan output ke nilai `InstanceType`.

```
aws ec2 describe-instance-types \
    --region us-east-1  \
    --filters Name=network-info.efa-supported,Values=true \
    --query "InstanceTypes[*].[InstanceType]"  \
    --output text | sort
```

## Sistem operasi yang didukung

Dukungan sistem operasi berbeda tergantung pada jenis prosesor. Tabel berikut menunjukkan sistem operasi yang didukung.

| Sistem operasi | Intel/AMD (**x86_64**) jenis contoh | AWS Jenis instance Graviton (**arm64**) |
|---|---|---|
| Amazon Linux 2023 | ✓ | ✓ |
| RHEL 8, 9, dan 10 | ✓ | ✓ |
| Debian 11, 12, dan 13 | ✓ | ✓ |
| Rocky Linux 8 dan 9 | ✓ | ✓ |
| Ubuntu 22.04, 24.04, dan 26.04 | ✓ | ✓ |
| SUSE Linux Enterprise 15 | ✓ | ✓ |

| Sistem operasi | Intel/AMD (**x86_64**) jenis contoh | AWS Jenis instance Graviton (**arm64**) |
| --- | --- | --- |
| SP2 dan yang lebih baru | | |

> **Note**
>
> Beberapa sistem operasi yang terdaftar mungkin tidak didukung dengan Intel MPI. Jika Anda menggunakan Intel MPI, lihat dokumentasi Intel MPI untuk memverifikasi dukungan untuk sistem operasi Anda.

## Batasan EFA

EFA memiliki batasan sebagai berikut:

- Penulisan RDMA tidak didukung dengan semua jenis instance. Untuk informasi selengkapnya, lihat Tipe instans yang didukung.
- Lalu lintas EFA [1] antara P4d/P4de/DL1 instance dan jenis instans lainnya saat ini tidak didukung.
- Tipe instans yang mendukung beberapa kartu jaringan dapat dikonfigurasi dengan satu EFA per kartu jaringan. Semua tipe instans yang didukung hanya mendukung satu EFA per instans.
- `c7g.16xlarge`,`m7g.16xlarge`, dan Inst `r7g.16xlarge` ans Khusus dan Host Khusus tidak didukung saat EFA dilampirkan.
- Lalu lintas EFA [1 tidak] dapat melintasi Zona Ketersediaan atau VPC. Ini tidak berlaku untuk lalu lintas IP normal dari perangkat ENA dari antarmuka EFA.
- Lalu lintas EFA [1 tidak dapat] dirutekan. Lalu lintas IP normal dari perangkat ENA dari antarmuka EFA tetap dapat dirutekan.
- EFA tidak didukung. AWS Outposts
- Perangkat EFA dari antarmuka EFA (EFA dengan ENA) didukung pada instans Windows hanya untuk aplikasi berbasis AWS Cloud Digital Interface Software Development Kit (AWS CDI SDK). Jika Anda melampirkan antarmuka EFA (EFA dengan ENA) ke instance Windows untuk aplikasi berbasis non-CDI SDK, itu berfungsi sebagai antarmuka ENA, tanpa kemampuan perangkat EFA tambahan. Antar EFA-only muka tidak didukung oleh aplikasi AWS CDI berbasis Windows

atau Linux. Untuk informasi selengkapnya, lihat Panduan AWS Cloud Digital Interface Pengguna Perangkat Pengembangan Perangkat Lunak (AWS CDI SDK).

[1] Lalu lintas EFA mengacu pada lalu lintas yang ditransmisikan melalui perangkat EFA baik EFA (EFA dengan ENA) atau antarmuka. EFA-only

## Harga EFA

EFA tersedia sebagai fitur jaringan Amazon EC2 opsional yang dapat Anda aktifkan pada instans apa pun yang didukung tanpa biaya tambahan.

## Memulai dengan EFA dan MPI untuk beban kerja HPC di Amazon EC2

Tutorial ini membantu Anda meluncurkan EFA dan cluster MPI-enabled instance untuk beban kerja HPC.

> **Note**
>
> `u7in-32tb.224xlarge`Inst `u7i-12tb.224xlarge` ans`u7in-16tb.224xlarge`,`u7in-24tb.224xlarge`,, dan dapat menjalankan hingga 128 proses MPI paralel dengan Open MPI atau hingga 256 proses MPI paralel dengan Intel MPI.

Tugas

- Langkah 1: Siapkan grup EFA-enabled keamanan
- Langkah 2: Luncurkan instans sementara
- Langkah 3: Instal perangkat lunak EFA
- Langkah 4: (Opsional) Aktifkan Open MPI 5
- Langkah 5: (Opsional) Instal Intel MPI
- Langkah 6: Menonaktifkan perlindungan ptrace
- Langkah 7. Konfirmasi instalasi
- Langkah 8: Menginstal aplikasi HPC Anda
- Langkah 9: Buat EFA-enabled AMI
- Langkah 10: Luncur EFA-enabled kan instance ke grup penempatan cluster
- Langkah 11: Mengakhiri instans sementara
- Langkah 12: Mengaktifkan SSH tanpa kata sandi

## Langkah 1: Siapkan grup EFA-enabled keamanan

EFA memerlukan grup keamanan yang memungkinkan semua lalu lintas masuk dan keluar ke dan dari grup keamanan itu sendiri. Prosedur berikut menciptakan grup keamanan yang memungkinkan semua lalu lintas masuk dan ke luar ke dan dari dirinya sendiri, dan yang memungkinkan lalu lintas SSH masuk dari alamat IPv4 apa pun untuk konektivitas SSH.

> **Important**
>
> Grup keamanan ini ditujukan untuk tujuan pengujian saja. Untuk lingkungan produksi Anda, kami sarankan Anda membuat aturan SSH masuk yang memungkinkan lalu lintas hanya dari alamat IP dari mana Anda terhubung, seperti alamat IP komputer Anda, atau berbagai alamat IP di jaringan lokal Anda.

Untuk skenario lainnya, lihat Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda.

Untuk membuat grup EFA-enabled keamanan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dalam panel navigasi, pilih Grup Keamanan, lalu pilih Buat grup keamanan.
3. Di jendela Buat Grup Keamanan, lakukan hal berikut:
   a. Untuk Nama grup keamanan, masukkan nama deskriptif untuk grup keamanan, seperti `EFA-enabled security group`.
   b. (Opsional) Untuk Deskripsi, masukkan deskripsi singkat grup keamanan.
   c. Untuk VPC, pilih VPC tempat Anda ingin meluncurkan instans Anda EFA-enabled .
   d. Pilih Buat grup keamanan.
4. Pilih grup keamanan yang Anda buat, dan pada tab Detail, salin ID grup keamanan.
5. Dengan grup keamanan yang masih dipilih, pilih Tindakan, Edit aturan masuk, lalu lakukan hal berikut:
   a. Pilih Tambahkan aturan.
   b. Untuk Tipe, pilih Semua lalu lintas.
   c. Untuk Tipe sumber, pilih Kustom dan tempelkan ID grup keamanan yang Anda salin ke dalam bidang.
   d. Pilih Tambahkan aturan.

e. Untuk Tipe, pilih SSH.

f. Untuk Jenis Sumber, pilih Anywhere-IPv4.

g. Pilih Simpan aturan.

6. Dengan grup keamanan yang masih dipilih, pilih Tindakan, Edit aturan keluar, lalu lakukan hal berikut:

a. Pilih Tambahkan aturan.

b. Untuk Tipe, pilih Semua lalu lintas.

c. Untuk Tipe tujuan, pilih Kustom dan tempelkan ID grup keamanan yang Anda salin ke dalam bidang.

d. Pilih Simpan aturan.

> **Important**
>
> Aturan masuk dan keluar yang merujuk sendiri (memungkinkan semua lalu lintas ke dan dari grup keamanan itu sendiri) wajib bagi EFA untuk berfungsi. Tanpa aturan ini, lalu lintas EFA antar instance akan diblokir.

## Langkah 2: Luncurkan instans sementara

Luncurkan instans sementara yang dapat Anda gunakan untuk menginstal dan mengonfigurasi komponen perangkat lunak EFA. Anda menggunakan instance ini untuk membuat EFA-enabled AMI dari mana Anda dapat meluncurkan EFA-enabled instans Anda.

Untuk meluncurkan instans sementara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih Luncurkan Instans untuk membuka wizard peluncuran instans baru.
3. (Opsional) Di bagian Nama dan tanda, berikan nama untuk instans, seperti `EFA-instance`. Nama ditetapkan ke instans sebagai tanda sumber daya (`Name=EFA-instance`).
4. Di bagian Application and OS Images, pilih AMI untuk salah satu sistem operasi yang didukung.
5. Di bagian Tipe instans, pilih tipe instans yang didukung.
6. Pada bagian Pasangan kunci, pilih pasangan kunci yang akan digunakan untuk instans.

7. Di bagian Pengaturan jaringan, pilih Edit, lalu lakukan hal berikut:

   a. Untuk Subnet, pilih subnet untuk meluncurkan instans.

   > **Important**
   >
   > Anda harus memilih subnet. Jika Anda tidak memilih subnet, Anda tidak dapat mengaktifkan instans untuk EFA.

   b. Untuk Firewall (grup keamanan), pilih Pilih grup keamanan yang ada, lalu pilih grup keamanan yang Anda buat di langkah sebelumnya.

   c. Perluas bagian Konfigurasi jaringan lanjutan.

   Untuk Antarmuka jaringan 1, pilih Indeks kartu jaringan = 0, indeks perangkat = 0, dan tipe antarmuka = EFA dengan ENA.

   (Opsional) Jika Anda menggunakan jenis instance multi-kartu, seperti `p4d.24xlarge` ataup`5.48xlarge`, untuk setiap antarmuka jaringan tambahan yang diperlukan, pilih Tam bah antarmuka jaringan, untuk Indeks kartu jaringan pilih indeks yang tidak digunakan berikutnya, lalu pilih Indeks perangkat = 1 dan Jenis antarmuka = EFA dengan ENA atau. EFA-only

8. Di bagian Penyimpanan, konfigurasikan volume sesuai kebutuhan.
9. Di panel Ringkasan di sebelah kanan, pilih Luncurkan instans.

> **Note**
>
> Pertimbangkan untuk mewajibkan penggunaan IMDSv2 untuk instance sementara serta AMI yang akan Anda buat di Langkah 9 kecuali Anda telah menetapkan IMDSv2 sebagai default untuk akun tersebut. Untuk informasi selengkapnya tentang langkah-langkah konfigurasi IMDSv2, lihat. Mengonfigurasi opsi metadata instans untuk instans baru

## Langkah 3: Instal perangkat lunak EFA

Instal EFA-enabled kernel, driver EFA, Libfabric, dan Open MPI stack yang diperlukan untuk mendukung EFA pada instans sementara Anda.

Langkah-langkahnya berbeda tergantung pada apakah Anda bermaksud untuk menggunakan EFA dengan Open MPI, Intel MPI, atau dengan Open MPI dan Intel MPI.

> **Note**
>
> Beberapa sistem operasi mungkin tidak didukung dengan Intel MPI. Jika Anda menggunakan Intel MPI, lihat dokumentasi Intel MPI untuk memverifikasi dukungan untuk sistem operasi Anda.

Untuk menginstal perangkat lunak EFA

1. Hubungkan ke instans yang Anda luncurkan. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..
2. Untuk memastikan bahwa semua paket perangkat lunak Anda telah diperbarui, lakukan pembaruan perangkat lunak cepat di instans Anda. Proses ini mungkin memerlukan waktu beberapa menit.

   - Amazon Linux 2023, RHEL 8/9, Rocky Linux 8/9

     ```
     $ sudo yum update -y
     ```

   - Ubuntu dan Debian

     ```
     $ sudo apt-get update && sudo apt-get upgrade -y
     ```

   - SUSE Linux Enterprise

     ```
     $ sudo zypper update -y
     ```

3. Boot ulang dan terhubung kembali ke instans Anda.
4. Unduh file penginstalan perangkat lunak EFA. File penginstalan perangkat lunak dikemas menjadi file tarball (`.tar.gz`) yang dikompresi. Untuk mengunduh versi stabil terbaru, gunakan perintah berikut.

   Anda juga bisa mendapatkan versi terbaru dengan mengganti nomor versi dengan `latest` dalam perintah sebelumnya.

   ```
   $ curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.50.0.tar.gz
   ```

5. (Opsional) Verifikasi keaslian dan integritas file tarball EFA (`.tar.gz`).

   Kami menyarankan Anda melakukan hal ini untuk memverifikasi identitas penerbit perangkat lunak dan memeriksa apakah file tersebut tidak diubah atau rusak sejak file tersebut diterbitkan. Jika Anda tidak ingin memverifikasi file tarball, lewati langkah ini.

   > **Note**
   >
   > Atau, jika Anda memilih untuk memverifikasi file tarball menggunakan checksum MD5 atau SHA256, lihat Memverifikasi penginstal EFA menggunakan checksum.

   a. Unduh kunci GPG publik dan impor ke dalam keyring Anda.

   ```
   $ wget https://efa-installer.amazonaws.com/aws-efa-installer.key && gpg --
   import aws-efa-installer.key
   ```

   Perintah tersebut harus mengembalikan nilai kunci. Catat nilai kunci, karena Anda membutuhkannya di langkah selanjutnya.

   b. Verifikasi sidik jari kunci GPG. Jalankan perintah berikut dan tentukan nilai kunci dari langkah sebelumnya.

   ```
   $ gpg --fingerprint key_value
   ```

   Perintah tersebut harus mengembalikan sidik jari yang identik dengan `4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC`. Jika sidik jari tidak cocok, jangan jalankan skrip instalasi EFA, dan hubungi Dukungan.

   c. Unduh file tanda tangan dan verifikasi tanda tangan pada file tarball EFA.

   ```
   $ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.50.0.tar.gz.sig
    && gpg --verify ./aws-efa-installer-1.50.0.tar.gz.sig
   ```

   Berikut ini adalah contoh output.

   ```
   gpg: Signature made Wed 29 Jul 2020 12:50:13 AM UTC using RSA key ID DD2D3CCC
   gpg: Good signature from "Amazon EC2 EFA <ec2-efa-maintainers@amazon.com>"
   gpg: WARNING: This key is not certified with a trusted signature!
   gpg:          There is no indication that the signature belongs to the owner.
   ```

```
Primary key fingerprint: 4E90 91BC BB97 A96B 26B1  5E59 A054 80B1 DD2D 3CCC
```

Jika hasilnya mencakup `Good signature`, dan sidik jari cocok dengan sidik jari yang dikembalikan di langkah sebelumnya, lanjutkan ke langkah berikutnya. Jika tidak, jangan jalankan skrip instalasi EFA, dan hubungi Dukungan.

6. Ekstraksi file dari file `.tar.gz` yang dikompresi dan navigasi ke dalam direktori yang diekstraksi.

```
$ tar -xf aws-efa-installer-1.50.0.tar.gz && cd aws-efa-installer
```

7. (Opsional) Verifikasi tanda tangan paket individual selama instalasi.

   Dimulai dengan EFA installer 1.48.0, installer menyertakan paket RPM dan DEB GPG-signed individual. Untuk memverifikasi keaslian dan integritas masing-masing paket selama instalasi, gunakan ben `--check-signatures` dera. Saat Anda mengaktifkan tanda ini, penginstal memverifikasi semua tanda tangan paket terlebih dahulu, dan hanya melanjutkan penginstalan jika setiap paket lulus verifikasi. Jika ada paket yang gagal verifikasi, penginstal segera keluar tanpa menginstal apa pun.

   a. Unduh kunci publik GPG.

   ```
   $ wget https://efa-installer.amazonaws.com/aws-efa-installer.key
   ```

   b. Ekspor jalur kunci. Kemudian, pada langkah berikutnya, tambahkan `--check-signatures` ke perintah instalasi dan gunakan alih- `sudo -E` alih `sudo` untuk melestarikan variabel lingkungan.

   ```
   $ export EFA_INSTALLER_KEY=$(pwd)/aws-efa-installer.key
   ```

   Pada RPM-based sistem (Amazon Linux, RHEL, Rocky Linux, dan SUSE), penginstal memverifikasi setiap RPM menggunakan. `rpm --checksig` Pada DEB-based sistem (Ubuntu, Debian), installer memverifikasi setiap DEB menggunakan verifikasi tanda tangan GPG.

   Jika verifikasi paket apa pun gagal, instalasi segera dibatalkan, melindungi sistem Anda dari paket yang rusak atau berbahaya.

> **Note**
>
> `--check-signatures`Bendera adalah opsional. Tanpa itu, penginstal tidak melakukan verifikasi tanda tangan individu.

8. Instal perangkat lunak EFA. Lakukan salah satu langkah berikut sesuai dengan kasus penggunaan Anda.

> **Note**
>
> Jika Anda menyelesaikan langkah opsional sebelumnya untuk mengatur verifikasi tanda tangan paket, tambahkan `--check-signatures` ke perintah instalasi dan gunakan al `sudo -E` ih-alih`sudo`. Sebagai contoh: `sudo -E ./efa_installer.sh -y --check-signatures`.

> **Note**
>
> EFA tidak mendukung NVIDIA GPUDirect dengan SUSE Linux. Jika menggunakan SUSE Linux, Anda juga harus menentukan opsi `--skip-kmod` untuk mencegah penginstalan kmod. Secara default, SUSE Linux tidak mengizinkan modul kernel out-of-tree.

Open MPI and Intel MPI

Jika bermaksud untuk menggunakan EFA dengan Open MPI dan MPI Intel, Anda harus menginstal perangkat lunak EFA dengan Libfabric dan Open MPI, serta harus menyelesaikan Langkah 5: Menginstal MPI Intel.

Untuk menginstal perangkat lunak EFA dengan Libfabric dan Open MPI, jalankan perintah berikut.

> **Note**
>
> Dari EFA 1.30.0, Open MPI 4.1 dan Open MPI 5 diinstal secara default. Anda dapat secara opsional menentukan versi Open MPI yang ingin diinstal. Untuk menginstal

hanya Open MPI 4.1, sertakan`--mpi=openmpi4`. Untuk menginstal Open MPI 5 saja, sertakan `--mpi=openmpi5`. Untuk menginstal keduanya, hilangkan opsi `--mpi`.

```
$ sudo ./efa_installer.sh -y
```

Libfabric diinstal ke `/opt/amazon/efa`. Open MPI 4.1 diinstal ke`/opt/amazon/openmpi`. Open MPI 5 diinstal ke `/opt/amazon/openmpi5`.

Open MPI only

Jika bermaksud untuk menggunakan EFA dengan Open MPI saja, Anda harus menginstal perangkat lunak EFA dengan Libfabric dan Open MPI. Anda juga dapat melewati Langkah 5: Menginstal MPI Intel. Untuk menginstal perangkat lunak EFA dengan Libfabric dan Open MPI, jalankan perintah berikut.

> **Note**
>
> Dari EFA 1.30.0, Open MPI 4.1 dan Open MPI 5 diinstal secara default. Anda dapat secara opsional menentukan versi Open MPI yang ingin diinstal. Untuk menginstal hanya Open MPI 4.1, sertakan`--mpi=openmpi4`. Untuk menginstal Open MPI 5 saja, sertakan `--mpi=openmpi5`. Untuk menginstal keduanya, hilangkan opsi `--mpi`.

```
$ sudo ./efa_installer.sh -y
```

Libfabric diinstal ke `/opt/amazon/efa`. Open MPI 4.1 diinstal ke`/opt/amazon/openmpi`. Open MPI 5 diinstal ke `/opt/amazon/openmpi5`.

Intel MPI only

Jika Anda berniat untuk menggunakan EFA dengan hanya Intel MPI, Anda harus menginstal perangkat lunak EFA tanpa Libfabric dan Open MPI. Dalam kasus ini, Intel MPI menggunakan Libfabric-nya yang tertanam. Jika memilih untuk melakukan ini, Anda harus menyelesaikan Langkah 5: Menginstal MPI Intel.

Untuk menginstal perangkat lunak EFA tanpa Libfabric dan Open MPI, jalankan perintah berikut.

```
$ sudo ./efa_installer.sh -y --minimal
```

9. Jika penginstal EFA meminta Anda untuk me-reboot instans, lakukanlah dan kemudian sambungkan kembali ke instans. Jika tidak, log out dari instans lalu log kembali untuk menyelesaikan penginstalan.
10. Hapus tarball yang tidak terkompresi dan tarball itu sendiri. Jika tidak, ini akan dimasukkan dalam EFA-enabled AMI yang Anda buat, meningkatkan ukurannya.

## Langkah 4: (Opsional) Aktifkan Open MPI 5

> **Note**
>
> Lakukan langkah ini hanya jika Anda bermaksud menggunakan Open MPI 5.

Dari EFA 1.30.0, Open MPI 4.1 dan Open MPI 5 diinstal secara default. Atau, Anda dapat memilih untuk menginstal hanya Open MPI 4.1 atau Open MPI 5.

Jika memilih untuk menginstal Open MPI 5 pada Langkah 3: Menginstal perangkat lunak EFA, dan bermaksud menggunakannya, Anda harus melakukan langkah-langkah berikut untuk mengaktifkannya.

Untuk mengaktifkan Open MPI 5

1. Tambahkan Open MPI 5 ke variabel lingkungan PATH.

   ```
   $ module load openmpi5
   ```

2. Verifikasi bahwa Open MPI 5 aktif untuk digunakan.

   ```
   $ which mpicc
   ```

   Perintah harus menampilkan direktori penginstalan Open MPI 5 - `/opt/amazon/openmpi5`.

3. (Opsional) Untuk memastikan bahwa Open MPI 5 ditambahkan ke variabel lingkungan PATH setiap kali instans dimulai, lakukan hal berikut:

bash shell

Tambahkan `module load openmpi5` ke `/home/username/.bashrc` dan `/home/username/.bash_profile`.

csh and tcsh shells

Tambahkan `module load openmpi5` ke `/home/username/.cshrc`.

Jika Anda perlu menghapus Open MPI 5 dari variabel lingkungan PATH, jalankan perintah berikut dan hapus perintah dari skrip startup shell.

```
$ module unload openmpi5
```

## Langkah 5: (Opsional) Instal Intel MPI

> **Important**
>
> Lakukan langkah ini hanya jika Anda ingin menggunakan Intel MPI. Jika Anda hanya ingin menggunakan Open MPI, lewati langkah ini.

Intel MPI membutuhkan instalasi tambahan dan konfigurasi variabel lingkungan.

Prasyarat

Pastikan bahwa pengguna yang melakukan langkah-langkah berikut ini memiliki izin sudo.

Untuk menginstal Intel MPI

1. Untuk mengunduh skrip penginstalan Intel MPI, lakukan hal berikut
   a. Kunjungi situs web Intel.
   b. Di bagian Perpustakaan Intel MPI di halaman web, pilih tautan untuk penginstal Intel MPI Library for Linux Offline.
2. Jalankan skrip penginstalan yang Anda unduh di langkah sebelumnya.

   ```
   $ sudo bash installation_script_name.sh
   ```

3. Di installer, pilih Accept & install.
4. Baca Program Peningkatan Intel, pilih opsi yang sesuai, lalu pilih Mulai Instalasi.
5. Saat instalasi selesai, pilih Tutup.
6. Secara default, Intel MPI menggunakan Libfabric (internal) yang disematkan. Anda dapat mengonfigurasi Intel MPI untuk menggunakan Libfabric yang dikirimkan dengan penginstal EFA sebagai gantinya. Biasanya, installer EFA dikirimkan dengan versi Libfabric yang lebih baru daripada Intel MPI. Dalam beberapa kasus, Libfabric yang dikirimkan dengan installer EFA lebih berperforma daripada Intel MPI. Untuk mengonfigurasi Intel MPI agar menggunakan Libfabric yang dikirimkan bersama penginstal EFA, lakukan salah satu hal berikut tergantung pada shell Anda.

   bash shells

   Tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

   ```
   export I_MPI_OFI_LIBRARY_INTERNAL=0
   ```

   csh and tcsh shells

   Tambahkan pernyataan berikut ke /home/*username*/.cshrc.

   ```
   setenv I_MPI_OFI_LIBRARY_INTERNAL 0
   ```

7. Tambahkan perintah source berikut ke skrip shell Anda untuk sumber skrip `vars.sh` dari direktori instalasi untuk mengatur lingkungan kompiler setiap kali instans dimulai. Lakukan salah satu langkah berikut sesuai dengan shell Anda.

   bash shells

   Tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

   ```
   source /opt/intel/oneapi/mpi/latest/env/vars.sh
   ```

   csh and tcsh shells

   Tambahkan pernyataan berikut ke /home/*username*/.cshrc.

```
source /opt/intel/oneapi/mpi/latest/env/vars.csh
```

8. Secara default, jika EFA tidak tersedia karena kesalahan konfigurasi, Intel MPI default ke tumpukan TCP/IP jaringan, yang dapat mengakibatkan kinerja aplikasi lebih lambat. Anda dapat mencegahnya dengan mengatur `I_MPI_OFI_PROVIDER` ke `efa`. Hal ini menyebabkan Intel MPI gagal dengan kesalahan berikut jika EFA tidak tersedia:

```
Abort (XXXXXX) on node 0 (rank 0 in comm 0): Fatal error in PMPI_Init: OtherMPI
 error,
MPIR_Init_thread (XXX).......:
MPID_Init (XXXX).............:
MPIDI_OFI_mpi_init_hook (XXXX):
open_fabric (XXXX)...........:
find_provider (XXXX).........:
OFI fi_getinfo() failed (ofi_init.c:2684:find_provider:
```

   Lakukan salah satu langkah berikut sesuai dengan shell Anda.

   bash shells

   Tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

   ```
   export I_MPI_OFI_PROVIDER=efa
   ```

   csh and tcsh shells

   Tambahkan pernyataan berikut ke /home/*username*/.cshrc.

   ```
   setenv I_MPI_OFI_PROVIDER efa
   ```

9. Secara default, Intel MPI tidak mencetak informasi debugging. Anda dapat menentukan tingkat verbositas yang berbeda untuk mengontrol informasi debugging. Nilai yang mungkin (dalam urutan jumlah detail yang mereka berikan) adalah: `0` (default), `1`, `2`, `3`, `4`, `5`. Level `1` dan lebih tinggi mencetak `libfabric version` dan `libfabric provider`. Gunakan `libfabric version` untuk memeriksa apakah Intel MPI menggunakan Libfabric internal atau Libfabric yang dikirimkan bersama penginstal EFA. Jika menggunakan Libfabric internal, versinya berakhiran dengan `impi`. Gunakan `libfabric provider` untuk memeriksa dengan Intel MPI

menggunakan EFA atau TCP/IP jaringan. Jika menggunakan EFA, nilainya adalah `efa`. Jika menggunakan TCP/IP, nilainya adalah`tcp;ofi_rxm`.

Untuk mengaktifkan informasi debugging, lakukan salah satu hal berikut sesuai dengan shell Anda.

bash shells

Tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

```
export I_MPI_DEBUG=value
```

csh and tcsh shells

Tambahkan pernyataan berikut ke /home/*username*/.cshrc.

```
setenv I_MPI_DEBUG value
```

10. Secara default, Intel MPI menggunakan memori bersama sistem operasi (`shm`) untuk komunikasi intra-node, dan menggunakan Libfabric (`ofi`) hanya untuk komunikasi antar node. Umumnya, konfigurasi ini memberikan performa terbaik. Namun, dalam beberapa kasus kain Intel MPI shm dapat menyebabkan aplikasi tertentu menggantung tanpa batas waktu.

    Untuk mengatasi masalah ini, Anda dapat memaksa Intel MPI untuk menggunakan Libfabric untuk komunikasi intra-simpul dan antar-simpul. Untuk melakukannya, lakukan salah satu langkah berikut sesuai dengan shell Anda.

    bash shells

    Tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

    ```
    export I_MPI_FABRICS=ofi
    ```

    csh and tcsh shells

    Tambahkan pernyataan berikut ke /home/*username*/.cshrc.

    ```
    setenv I_MPI_FABRICS ofi
    ```

> **Note**
>
> Penyedia Libfabric EFA menggunakan memori bersama sistem operasi untuk komunikasi intra-simpul. Ini berarti bahwa pengaturan `I_MPI_FABRICS` untuk `ofi` menghasilkan performa yang mirip dengan konfigurasi `shm:ofi` default.

11. Keluar dari instans lalu masuk kembali.

Jika Anda tidak ingin menggunakan Intel MPI lagi, hapus variabel lingkungan dari skrip startup shell.

## Langkah 6: Menonaktifkan perlindungan ptrace

Untuk meningkatkan performa aplikasi HPC Anda, Libfabric menggunakan memori lokal instans untuk komunikasi antarproses ketika proses berjalan pada instans yang sama.

Fitur memori bersama menggunakan Cross Memory Attach (CMA), yang tidak didukung oleh perlindungan ptrace. Jika Anda menggunakan distribusi Linux yang memiliki perlindungan ptrace yang diaktifkan secara default, seperti Ubuntu, Anda harus menonaktifkannya. Jika distribusi Linux Anda tidak memiliki perlindungan ptrace yang diaktifkan secara default, lewati langkah ini.

Untuk menonaktifkan perlindungan ptrace

Lakukan salah satu langkah berikut:

- Untuk menonaktifkan sementara perlindungan ptrace untuk tujuan pengujian, jalankan perintah berikut.

```
$ sudo sysctl -w kernel.yama.ptrace_scope=0
```

- Untuk menonaktifkan perlindungan ptrace secara permanen, tambahkan `kernel.yama.ptrace_scope = 0` ke `/etc/sysctl.d/10-ptrace.conf` dan nyalakan ulang instans.

## Langkah 7. Konfirmasi instalasi

Untuk mengkonfirmasi instalasi yang berhasil

1. Untuk mengonfirmasi bahwa MPI berhasil diinstal, jalankan perintah berikut:

```
$ which mpicc
```

- Untuk Open MPI, jalur yang dikembalikan harus menyertakan `/opt/amazon/`
- Untuk Intel MPI, jalur yang dikembalikan harus menyertakan `/opt/intel/`. Jika Anda tidak mendapatkan output yang diharapkan, pastikan Anda telah mendapatkan skrip Intel MPI `vars.sh`.

2. Untuk mengonfirmasi bahwa komponen perangkat lunak EFA dan Libfabric berhasil diinstal, jalankan perintah berikut.

```
$ fi_info -p efa -t FI_EP_RDM
```

Perintah tersebut harus mengembalikan informasi tentang antarmuka Libfabric EFA. Contoh berikut menunjukkan output perintah.

```
provider: efa
    fabric: EFA-fe80::94:3dff:fe89:1b70
    domain: efa_0-rdm
    version: 2.0
    type: FI_EP_RDM
    protocol: FI_PROTO_EFA
```

## Langkah 8: Menginstal aplikasi HPC Anda

Instal aplikasi HPC di instans sementara. Prosedur penginstalan bervariasi tergantung pada aplikasi HPC tertentu. Untuk informasi selengkapnya, lihat Mengel ola pembaruan OS di Panduan Pengguna  Amazon Linux 2023.

> **Note**
>
> Lihat dokumentasi aplikasi HPC Anda untuk petunjuk penginstalan.

## Langkah 9: Buat EFA-enabled AMI

Setelah menginstal komponen perangkat lunak yang diperlukan, Anda membuat AMI yang dapat Anda gunakan kembali untuk meluncurkan EFA-enabled instans Anda.

Untuk membuat AMI dari instans sementara Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans sementara yang Anda buat dan pilih Tindakan, Gambar, Buat gambar.
4. Untuk Buat gambar, lakukan hal berikut:
   a. Untuk Nama gambar, masukkan nama deskriptif untuk AMI.
   b. (Opsional) Untuk Deskripsi gambar, masukkan deskripsi singkat tentang tujuan AMI.
   c. Pilih Buat gambar.
5. Di panel navigasi, pilih AMI.
6. Temukan AMI yang Anda buat dalam daftar. Tunggu hingga status berubah dari `pending` menjadi `available` sebelum melanjutkan ke langkah berikutnya.

## Langkah 10: Luncur EFA-enabled kan instance ke grup penempatan cluster

Luncur EFA-enabled kan instans Anda ke grup penempatan cluster menggunakan EFA-enabled AMI yang Anda buat di Langkah 7, dan grup EFA-enabled keamanan yang Anda buat di Langkah 1.

> **Note**
>
> - Ini bukan persyaratan mutlak untuk meluncurkan EFA-enabled instans Anda ke grup penempatan cluster. Namun, sebaiknya jalankan EFA-enabled instans Anda di grup penempatan cluster karena meluncurkan instans ke grup latensi rendah di Zona Ketersediaan tunggal.
> - Untuk memastikan kapasitas tersedia saat Anda menskalakan instans cluster, Anda dapat membuat Reservasi Kapasitas untuk grup penempatan cluster Anda. Untuk informasi selengkapnya, lihat Gunakan Reservasi Kapasitas dengan grup penempatan.

Untuk meluncurkan sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih Luncurkan Instans untuk membuka wizard peluncuran instans baru.

3. (Opsional) Di bagian Nama dan tanda, berikan nama untuk instans, seperti `EFA-instance`. Nama ditetapkan ke instans sebagai tanda sumber daya (`Name=`*`EFA-instance`*).
4. Di bagian Application and OS Images, pilih AMI Saya, lalu pilih AMI yang Anda buat pada langkah sebelumnya.
5. Di bagian Tipe instans, pilih tipe instans yang didukung.
6. Pada bagian Pasangan kunci, pilih pasangan kunci yang akan digunakan untuk instans.
7. Di bagian Pengaturan jaringan, pilih Edit, lalu lakukan hal berikut:

   a. Untuk Subnet, pilih subnet untuk meluncurkan instans.

   > **Important**
   >
   > Anda harus memilih subnet. Jika Anda tidak memilih subnet, Anda tidak dapat mengaktifkan instans untuk EFA.

   b. Untuk Firewall (grup keamanan), pilih Pilih grup keamanan yang ada, lalu pilih grup keamanan yang Anda buat di langkah sebelumnya.

   c. Perluas bagian Konfigurasi jaringan lanjutan.

   Untuk Antarmuka jaringan 1, pilih Indeks kartu jaringan = 0, indeks perangkat = 0, dan tipe antarmuka = EFA dengan ENA.

   (Opsional) Jika Anda menggunakan jenis instance multi-kartu, seperti `p4d.24xlarge` ataup`5.48xlarge`, untuk setiap antarmuka jaringan tambahan yang diperlukan, pilih Tam bah antarmuka jaringan, untuk Indeks kartu jaringan pilih indeks yang tidak digunakan berikutnya, lalu pilih Indeks perangkat = 1 dan Jenis antarmuka = EFA dengan ENA atau. EFA-only
8. (Opsional) Di bagian Penyimpanan, konfigurasikan volume sesuai kebutuhan.
9. Di bagian Detail lanjutan, untuk nama grup Penempatan, pilih grup penempatan klaster untuk meluncurkan instans. Jika Anda perlu membuat grup penempatan klaster baru, pilih Buat grup penempatan baru.
10. Di panel Ringkasan di sebelah kanan, untuk Jumlah instance, masukkan jumlah EFA-enabled instance yang ingin Anda luncurkan, lalu pilih Luncurkan instance.

## Langkah 11: Mengakhiri instans sementara

Pada titik ini, Anda tidak lagi memerlukan instance yang Anda luncurkan di Langkah 2. Anda dapat mengakhiri instans agar biaya tidak dibebankan lagi padanya.

Untuk mengakhiri instans sementara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance sementara yang Anda buat, lalu pilih Ac tions, Instance state, Hentikan (hapus) instance.
4. Saat diminta konfirmasi, pilih H entikan (hapus).

## Langkah 12: Mengaktifkan SSH tanpa kata sandi

Agar aplikasi Anda dapat berjalan di semua instans dalam klaster, Anda harus mengaktifkan akses SSH tanpa kata sandi dari simpul pemimpin ke simpul anggota. Simpul pemimpin adalah instans dari mana Anda menjalankan aplikasi Anda. Instans yang tersisa di klaster adalah simpul anggota.

Untuk mengaktifkan SSH tanpa kata sandi antar instans dalam klaster

1. Pilih satu instans dalam klaster sebagai simpul pemimpin, dan hubungkan ke instans tersebut.
2. Nonaktifkan `strictHostKeyChecking` dan aktifkan `ForwardAgent` pada simpul pemimpin. Buka `~/.ssh/config` menggunakan editor teks pilihan Anda dan tambahkan berikut ini.

```
Host *
    ForwardAgent yes
Host *
    StrictHostKeyChecking no
```

3. Membuat pasangan kunci RSA.

```
$ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
```

   Pasangan kunci dibuat dalam direktori `$HOME/.ssh/`.

4. Ubah izin kunci privat pada simpul pemimpin.

```
$ chmod 600 ~/.ssh/id_rsa
```

```
chmod 600 ~/.ssh/config
```

5. Buka `~/.ssh/id_rsa.pub` menggunakan editor teks pilihan Anda dan salin kunci.
6. Untuk setiap simpul anggota di klaster, lakukan hal berikut:
   a. Hubungkan dengan instans.
   b. Buka `~/.ssh/authorized_keys` menggunakan editor teks pilihan Anda tambahkan kunci publik yang Anda salin sebelumnya.
7. Untuk menguji apakah SSH tanpa kata sandi berfungsi seperti yang diharapkan, hubungkan ke simpul pemimpin Anda dan jalankan perintah berikut.

   ```
   $ ssh member_node_private_ip
   ```

   Anda harus terhubung ke simpul anggota tanpa diminta untuk memasukkan kunci atau kata sandi.

## Memulai dengan EFA dan NCCL untuk beban kerja ML di Amazon EC2

NVIDIA Collective Communications Library (NCCL) adalah perpustakaan rutinitas komunikasi kolektif standar untuk banyak GPU pada simpul tunggal atau beberapa simpul. Anda dapat menggunakan NCCL bersama dengan EFA, Libfabric, dan MPI untuk mendukung berbagai beban kerja pembelajaran mesin. Untuk informasi lebih lanjut, lihat situs web NCCL.

Persyaratan

- Jenis instans yang didukung termasuk tipe instance seri EFA-supported P dan seri G. Untuk informasi selengkapnya, lihat Inst ans komputasi akselerasi Amazon EC2.
- AMI dasar yang didukung: Amazon Linux 2023, Ubuntu 26.04, Ubuntu 24.04, Ubuntu 22.04, Debian 12 dan RHEL 10.
- EFA hanya mendukung NCCL 2.4.2 dan yang lebih baru.

Untuk informasi selengkapnya tentang menjalankan beban kerja pembelajaran mesin dengan EFA dan NCCL menggunakan AWS Deep Learning AMIs, lihat Menggunakan EFA di D LAMI di Panduan Pengembang. AWS Deep Learning AMIs

Langkah-langkah

- Langkah 1: Siapkan grup EFA-enabled keamanan

- Langkah 2: Luncurkan instans sementara
- Langkah 3: Instal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN
- Langkah 4: Instal GDRCopy
- Langkah 5: Instal perangkat lunak EFA
- Langkah 6: Instal NCCL
- Langkah 7: Instal uji NCCL
- Langkah 8: Uji konfigurasi EFA dan NCCL Anda
- Langkah 9: Instal aplikasi machine learning Anda
- Langkah 10: Buat EFA dan AMI NCCL-enabled
- Langkah 11: Mengakhiri instans sementara
- Langkah 12: Luncurkan EFA dan NCCL-enabled instance ke grup penempatan cluster
- Langkah 13: Mengaktifkan SSH tanpa kata sandi


## Langkah 1: Siapkan grup EFA-enabled keamanan

EFA memerlukan grup keamanan yang memungkinkan semua lalu lintas masuk dan keluar ke dan dari grup keamanan itu sendiri. Prosedur berikut menciptakan grup keamanan yang memungkinkan semua lalu lintas masuk dan ke luar ke dan dari dirinya sendiri, dan yang memungkinkan lalu lintas SSH masuk dari alamat IPv4 apa pun untuk konektivitas SSH.

**Important**

- Grup keamanan ini ditujukan untuk tujuan pengujian saja. Untuk lingkungan produksi Anda, kami sarankan Anda membuat aturan SSH masuk yang memungkinkan lalu lintas hanya dari alamat IP dari mana Anda terhubung, seperti alamat IP komputer Anda, atau berbagai alamat IP di jaringan lokal Anda.
- Aturan masuk dan keluar yang merujuk sendiri (memungkinkan semua lalu lintas ke dan dari grup keamanan itu sendiri) wajib bagi EFA untuk berfungsi. Tanpa aturan ini, lalu lintas EFA antar instance akan diblokir dan komunikasi NCCL akan gagal.

Untuk skenario lainnya, lihat Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda.

Untuk membuat grup EFA-enabled keamanan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dalam panel navigasi, pilih Grup Keamanan, lalu pilih Buat grup keamanan.
3. Di jendela Buat Grup Keamanan, lakukan hal berikut:
   a. Untuk Nama grup keamanan, masukkan nama deskriptif untuk grup keamanan, seperti `EFA-enabled security group`.
   b. (Opsional) Untuk Deskripsi, masukkan deskripsi singkat grup keamanan.
   c. Untuk VPC, pilih VPC tempat Anda ingin meluncurkan instans Anda EFA-enabled .
   d. Pilih Buat grup keamanan.
4. Pilih grup keamanan yang Anda buat, dan pada tab Detail, salin ID grup keamanan.
5. Dengan grup keamanan yang masih dipilih, pilih Tindakan, Edit aturan masuk, lalu lakukan hal berikut:
   a. Pilih Tambahkan aturan.
   b. Untuk Tipe, pilih Semua lalu lintas.
   c. Untuk Tipe sumber, pilih Kustom dan tempelkan ID grup keamanan yang Anda salin ke dalam bidang.
   d. Pilih Tambahkan aturan.
   e. Untuk Tipe, pilih SSH.
   f. Untuk Jenis Sumber, pilih Anywhere-IPv4.
   g. Pilih Simpan aturan.
6. Dengan grup keamanan yang masih dipilih, pilih Tindakan, Edit aturan keluar, lalu lakukan hal berikut:
   a. Pilih Tambahkan aturan.
   b. Untuk Tipe, pilih Semua lalu lintas.
   c. Untuk Tipe tujuan, pilih Kustom dan tempelkan ID grup keamanan yang Anda salin ke dalam bidang.
   d. Pilih Simpan aturan.

## Langkah 2: Luncurkan instans sementara

Luncurkan instans sementara yang dapat Anda gunakan untuk menginstal dan mengonfigurasi komponen perangkat lunak EFA. Anda menggunakan instance ini untuk membuat EFA-enabled AMI dari mana Anda dapat meluncurkan EFA-enabled instans Anda.

Untuk meluncurkan instans sementara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih Luncurkan Instans untuk membuka wizard peluncuran instans baru.
3. (Opsional) Di bagian Nama dan tanda, berikan nama untuk instans, seperti `EFA-instance`. Nama ditetapkan ke instans sebagai tanda sumber daya (`Name=EFA-instance`).
4. Di bagian Aplikasi dan Gambar OS, pilih AMI untuk salah satu sistem operasi yang didukung.
5. Di bagian Jenis Instance, pilih jenis instance yang didukung.
6. Pada bagian Pasangan kunci, pilih pasangan kunci yang akan digunakan untuk instans.
7. Di bagian Pengaturan jaringan, pilih Edit, lalu lakukan hal berikut:

   a. Untuk Subnet, pilih subnet untuk meluncurkan instans.

   > **Important**
   >
   > Anda harus memilih subnet. Jika Anda tidak memilih subnet, Anda tidak dapat mengaktifkan instans untuk EFA.

   b. Untuk Firewall (grup keamanan), pilih Pilih grup keamanan yang ada, lalu pilih grup keamanan yang Anda buat di langkah sebelumnya.

   c. Perluas bagian Konfigurasi jaringan lanjutan.

   Untuk Antarmuka jaringan 1, pilih Indeks kartu jaringan = 0, indeks perangkat = 0, dan tipe antarmuka = EFA dengan ENA.

   (Opsional) Jika Anda menggunakan jenis instance multi-kartu, untuk setiap antarmuka jaringan tambahan yang diperlukan, pilih Tambahkan antarmuka jaringan, untuk Indeks kartu jaringan pilih indeks yang tidak digunakan berikutnya, lalu pilih Indeks perangkat = 1 dan Jenis antarmuka = EFA dengan ENA atau. EFA-only
8. Di bagian Penyimpanan, konfigurasikan volume sesuai kebutuhan.

> **Note**
>
> Anda harus menyediakan penyimpanan tambahan 10 hingga 20 GiB untuk NVIDIA CUDA Toolkit. Jika Anda tidak menyediakan penyimpanan yang cukup, Anda akan menerima `insufficient disk space` kesalahan saat mencoba menginstal driver NVIDIA dan toolkit CUDA.

9. Di panel Ringkasan di sebelah kanan, pilih Luncurkan instans.

## Langkah 3: Instal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN

Amazon Linux 2023

Untuk menginstal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN

1. Untuk memastikan bahwa semua paket perangkat lunak Anda telah diperbarui, lakukan pembaruan perangkat lunak cepat di instans Anda.

```
$ sudo dnf upgrade -y && sudo reboot
```

   Hubungkan kembali ke instans Anda setelah boot ulang.

2. Instal utilitas yang diperlukan untuk menginstal driver GPU NVIDIA dan NVIDIA CUDA Toolkit.

```
$ sudo dnf groupinstall 'Development Tools' -y && sudo dnf install -y dkms
 kernel-devel-$(uname -r) kernel-headers-$(uname -r)
```

3. Nonaktifkan driver `nouveau` open source.

   a. Instal utilitas yang diperlukan dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
$ sudo yum install -y wget kernel-devel-$(uname -r) kernel-headers-$(uname -
r)
```

   b. Tambahkan `nouveau` ke file `/etc/modprobe.d/blacklist.conf` daftar tolak.

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
```

```
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

c. Tambahkan `GRUB_CMDLINE_LINUX="rdblacklist=nouveau"` ke `grub` file dan bangun kembali konfigurasi GRUB.

```
$ echo 'GRUB_CMDLINE_LINUX="rdblacklist=nouveau"' | sudo tee -a /etc/
default/grub \
&& sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

4. Boot ulang dan terhubung kembali ke instans Anda.
5. Tambahkan repositori jaringan CUDA.

```
$ sudo yum-config-manager --add-repo https://developer.download.nvidia.com/
compute/cuda/repos/rhel8/x86_64/cuda-rhel8.repo
```

6. Unduh dan instal driver GPU NVIDIA.

```
$ wget https://us.download.nvidia.com/tesla/580.167.08/NVIDIA-Linux-
x86_64-580.167.08.run \
&& sudo sh NVIDIA-Linux-x86_64-580.167.08.run -m kernel-open --no-drm --disable-
nouveau --dkms --silent
```

7. Instal NVIDIA CUDA Toolkit dan cuDNN.

```
$ sudo dnf install -y cuda-toolkit-13-0 libcudnn9-cuda-13 libcudnn9-devel-
cuda-13
```

8. Boot ulang dan terhubung kembali ke instans Anda.
9. (Instans dengan NVSwitch, seperti inst P-series ans multi-GPU) Instal dan mulai NVIDIA Fabric Manager. G-series instance tidak menggunakan NVSwitch dan tidak memerlukan Fabric Manager.

```
$ sudo dnf install -y https://developer.download.nvidia.com/compute/cuda/repos/
rhel8/x86_64/nvidia-fabricmanager-580.167.08-1.el8.x86_64.rpm \
&& sudo systemctl enable nvidia-fabricmanager && sudo systemctl start nvidia-
fabricmanager
```

10. Pastikan bahwa jalur CUDA diatur setiap kali instans dimulai.

    - Untuk shell bash, tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

      ```
      export PATH=/usr/local/cuda/bin:$PATH
      export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
      lib64:$LD_LIBRARY_PATH
      ```

    - Untuk shell tcsh, tambahkan pernyataan berikut ke /home/*username*/.cshrc.

      ```
      setenv PATH=/usr/local/cuda/bin:$PATH
      setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
      lib64:$LD_LIBRARY_PATH
      ```

11. Untuk mengonfirmasi bahwa driver GPU NVIDIA berfungsi, jalankan perintah berikut.

    ```
    $ nvidia-smi -q | head
    ```

    Perintah harus mengembalikan informasi tentang GPU NVIDIA, driver GPU NVIDIA, dan NVIDIA CUDA Toolkit.

Ubuntu 26.04, Ubuntu 24.04, and Ubuntu 22.04

Untuk menginstal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN

1. Untuk memastikan bahwa semua paket perangkat lunak Anda telah diperbarui, lakukan pembaruan perangkat lunak cepat di instans Anda.

   ```
   $ sudo apt-get update && sudo apt-get upgrade -y
   ```

2. Instal utilitas yang diperlukan untuk menginstal driver GPU NVIDIA dan NVIDIA CUDA Toolkit.

   ```
   $ sudo apt-get update && sudo apt-get install build-essential -y
   ```

3. Untuk menggunakan driver GPU NVIDIA, Anda harus terlebih dahulu menonaktifkan driver nouveau open source.

a. Instal utilitas yang diperlukan dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

```
$ sudo apt-get install -y gcc make linux-headers-$(uname -r)
```

b. Tambahkan nouveau ke file `/etc/modprobe.d/blacklist.conf` daftar tolak.

```
$ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
blacklist vga16fb
blacklist nouveau
blacklist rivafb
blacklist nvidiafb
blacklist rivatv
EOF
```

c. Buka `/etc/default/grub` menggunakan editor teks pilihan Anda dan tambahkan berikut ini.

```
GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
```

d. Bangun kembali konfigurasi GRUB.

```
$ sudo update-grub
```

4. Boot ulang dan terhubung kembali ke instans Anda.
5. Tambahkan repositori CUDA dan instal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN.

   - Ubuntu 26.04

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2604/
x86_64/cuda-keyring_1.1-1_all.deb \
&& sudo dpkg -i cuda-keyring_1.1-1_all.deb \
&& sudo add-apt-repository -y 'deb [trusted=yes] https://
developer.download.nvidia.com/compute/cuda/repos/ubuntu2604/x86_64 /' \
&& sudo apt-get update
```

```
$ sudo apt-get install -y nvidia-open cuda-toolkit-13-3 libcudnn9-cuda-13
 libcudnn9-dev-cuda-13
```

- Ubuntu 24.04

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2404/
x86_64/cuda-keyring_1.1-1_all.deb \
&& sudo dpkg -i cuda-keyring_1.1-1_all.deb \
&& sudo add-apt-repository -y 'deb [trusted=yes] https://
developer.download.nvidia.com/compute/cuda/repos/ubuntu2404/x86_64 /' \
&& sudo apt-get update
```

```
$ sudo apt-get install -y nvidia-open-580 cuda-toolkit-13-0 libcudnn9-cuda-13
 libcudnn9-dev-cuda-13
```

- Ubuntu 22.04

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/
x86_64/cuda-keyring_1.1-1_all.deb \
&& sudo dpkg -i cuda-keyring_1.1-1_all.deb \
&& sudo DEBIAN_FRONTEND=noninteractive add-apt-repository -y "deb https://
developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86_64/ /" \
&& sudo apt-get update
```

```
$ sudo apt-get install -y nvidia-open-580 cuda-toolkit-13-0 libcudnn9-cuda-13
 libcudnn9-dev-cuda-13
```

6. Boot ulang dan terhubung kembali ke instans Anda.
7. (Instans dengan NVSwitch, seperti inst P-series ans multi-GPU) Instal dan mulai NVIDIA Fabric Manager. G-series instance tidak menggunakan NVSwitch dan tidak memerlukan Fabric Manager.

   - Ubuntu 26.04

   ```
   $ sudo apt-get install -y nvidia-fabricmanager \
   && sudo systemctl enable nvidia-fabricmanager && sudo systemctl start nvidia-
   fabricmanager
   ```

   - Ubuntu 24.04 dan Ubuntu 22.04

   ```
   $ sudo apt-get install -y nvidia-fabricmanager-580 \
   && sudo systemctl enable nvidia-fabricmanager && sudo systemctl start nvidia-
   fabricmanager
   ```

8. Pastikan bahwa jalur CUDA diatur setiap kali instans dimulai.

   - Untuk shell bash, tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

     ```
     export PATH=/usr/local/cuda/bin:$PATH
     export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

   - Untuk shell tcsh, tambahkan pernyataan berikut ke /home/*username*/.cshrc.

     ```
     setenv PATH=/usr/local/cuda/bin:$PATH
     setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

9. Untuk mengonfirmasi bahwa driver GPU NVIDIA berfungsi, jalankan perintah berikut.

   ```
   $ nvidia-smi -q | head
   ```

   Perintah harus mengembalikan informasi tentang GPU NVIDIA, driver GPU NVIDIA, dan NVIDIA CUDA Toolkit.

Debian 12

Untuk menginstal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN

1. Instal utilitas yang diperlukan untuk menginstal driver GPU NVIDIA dan NVIDIA CUDA Toolkit.

   ```
   $ sudo apt-get install -y build-essential gcc make linux-headers-$(uname -r)
    dkms
   ```

2. Nonaktifkan driver nouveau open source.

   ```
   $ sudo sed -i 's/GRUB_CMDLINE_LINUX=""/
   GRUB_CMDLINE_LINUX="rdblacklist=nouveau"/' /etc/default/grub && sudo update-grub
   ```

3. Boot ulang dan terhubung kembali ke instans Anda.
4. Tambahkan repositori CUDA dan instal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN.

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/debian12/x86_64/
cuda-keyring_1.1-1_all.deb \
&& sudo dpkg -i cuda-keyring_1.1-1_all.deb \
&& sudo DEBIAN_FRONTEND=noninteractive add-apt-repository -y "deb https://
developer.download.nvidia.com/compute/cuda/repos/debian12/x86_64/ /" \
&& sudo apt-get update
```

```
$ sudo apt-get install -y nvidia-open cuda-toolkit-13-0 libcudnn9-cuda-13
 libcudnn9-dev-cuda-13
```

5. Konfigurasikan modul kernel NVIDIA UVM.

```
$ uvm_ko=$(find /lib/modules/$(uname -r) -name 'nvidia*uvm*.ko*' 2>/dev/null
 | head -1) && if [ -n "$uvm_ko" ]; then real=$(basename "$uvm_ko"); real=
${real%.ko*}; echo "$real" | sudo tee /etc/modules-load.d/nvidia-uvm.conf; if
 [ "$real" != "nvidia-uvm" ]; then echo "alias nvidia-uvm $real" | sudo tee /
etc/modprobe.d/nvidia-uvm.conf; fi; sudo modprobe "$real" || true; fi && sudo
 modprobe nvidia
```

6. Boot ulang dan terhubung kembali ke instans Anda.
7. (Instans dengan NVSwitch, seperti inst P-series ans multi-GPU) Instal dan mulai NVIDIA Fabric Manager. G-series instance tidak menggunakan NVSwitch dan tidak memerlukan Fabric Manager.

   a. Tentukan versi modul kernel NVIDIA.

   ```
   $ cat /proc/driver/nvidia/version | grep "Kernel Module"
   ```

   Berikut ini adalah output contoh.

   ```
   NVRM version: NVIDIA UNIX x86_64 Kernel Module  610.43.02  ...
   ```

   Dalam contoh sebelumnya, versi utama `610` dari modul kernel diinstal.

   b. Instal NVIDIA Fabric Manager menggunakan versi utama yang diidentifikasi pada langkah sebelumnya.

   ```
   $ sudo apt-get install -y nvidia-fabricmanager-major_version_number \
   ```

```
&& sudo systemctl enable nvidia-fabricmanager && sudo systemctl start
 nvidia-fabricmanager
```

8. Pastikan bahwa jalur CUDA diatur setiap kali instans dimulai.

   - Untuk shell bash, tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

     ```
     export PATH=/usr/local/cuda/bin:$PATH
     export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

   - Untuk shell tcsh, tambahkan pernyataan berikut ke /home/*username*/.cshrc.

     ```
     setenv PATH=/usr/local/cuda/bin:$PATH
     setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

9. Untuk mengonfirmasi bahwa driver GPU NVIDIA berfungsi, jalankan perintah berikut.

   ```
   $ nvidia-smi -q | head
   ```

   Perintah harus mengembalikan informasi tentang GPU NVIDIA, driver GPU NVIDIA, dan NVIDIA CUDA Toolkit.

RHEL 10

Untuk menginstal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN

1. Untuk memastikan bahwa semua paket perangkat lunak Anda telah diperbarui, lakukan pembaruan perangkat lunak cepat di instans Anda.

   ```
   $ sudo dnf upgrade -y && sudo reboot
   ```

   Hubungkan kembali ke instans Anda setelah boot ulang.

2. Instal utilitas yang diperlukan untuk menginstal driver GPU NVIDIA dan NVIDIA CUDA Toolkit.

   ```
   $ sudo dnf groupinstall 'Development Tools' -y \
   && sudo dnf install -y dkms kernel-devel-$(uname -r) \
   ```

```
&& sudo dnf install -y https://dl.fedoraproject.org/pub/epel/epel-release-
latest-10.noarch.rpm
```

3. Nonaktifkan driver nouveau open source.

   a. Tambahkan nouveau ke file /etc/modprobe.d/blacklist.conf daftar tolak.

   ```
   $ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
   blacklist vga16fb
   blacklist nouveau
   blacklist rivafb
   blacklist nvidiafb
   blacklist rivatv
   EOF
   ```

   b. Tambahkan GRUB_CMDLINE_LINUX="rdblacklist=nouveau" ke grub file dan bangun kembali konfigurasi GRUB.

   ```
   $ echo 'GRUB_CMDLINE_LINUX="rdblacklist=nouveau"' | sudo tee -a /etc/
   default/grub \
   && sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

4. Boot ulang dan terhubung kembali ke instans Anda.
5. Tambahkan repositori CUDA dan instal driver GPU NVIDIA, NVIDIA CUDA Toolkit, dan cuDNN.

   ```
   $ sudo yum-config-manager --add-repo https://developer.download.nvidia.com/
   compute/cuda/repos/rhel10/x86_64/cuda-rhel10.repo \
   && sudo dnf install -y nvidia-open-580.167.08 cuda-toolkit-13-0 libcudnn9-
   cuda-13 libcudnn9-devel-cuda-13
   ```

6. Boot ulang dan terhubung kembali ke instans Anda.
7. (Instans dengan NVSwitch, seperti inst P-series ans multi-GPU) Instal dan mulai NVIDIA Fabric Manager. G-series instance tidak menggunakan NVSwitch dan tidak memerlukan Fabric Manager.

   ```
   $ sudo dnf install -y https://developer.download.nvidia.com/compute/cuda/repos/
   rhel10/x86_64/nvidia-fabricmanager-580.167.08-1.x86_64.rpm \
   && sudo systemctl enable nvidia-fabricmanager && sudo systemctl start nvidia-
   fabricmanager
   ```

8. Pastikan bahwa jalur CUDA diatur setiap kali instans dimulai.

   - Untuk shell bash, tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

     ```
     export PATH=/usr/local/cuda/bin:$PATH
     export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

   - Untuk shell tcsh, tambahkan pernyataan berikut ke /home/*username*/.cshrc.

     ```
     setenv PATH=/usr/local/cuda/bin:$PATH
     setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/
     lib64:$LD_LIBRARY_PATH
     ```

9. Untuk mengonfirmasi bahwa driver GPU NVIDIA berfungsi, jalankan perintah berikut.

   ```
   $ nvidia-smi -q | head
   ```

   Perintah harus mengembalikan informasi tentang GPU NVIDIA, driver GPU NVIDIA, dan NVIDIA CUDA Toolkit.

## Langkah 4: Instal GDRCopy

Instal GDRCopy untuk meningkatkan performa Libfabric. Untuk informasi selengkapnya tentang GDRCopy, lihat repositori GDRCopy.

Amazon Linux 2023

Untuk menginstal GDRCopy

1. Instal dependensi yang diperlukan.

   ```
   $ sudo yum -y install dkms rpm-build make check check-devel
   ```

2. Mengunduh dan mengekstraksi paket GDRcopy.

   ```
   $ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.5.2.tar.gz \
   && tar xf v2.5.2.tar.gz && cd gdrcopy-2.5.2/packages
   ```

3. Membangun paket RPM GDRCopy.

```
$ CUDA=/usr/local/cuda ./build-rpm-packages.sh
```

4. Instal paket RPM GDRCopy.

```
$ sudo rpm -Uvh gdrcopy-kmod-2.5.2*dkms*.rpm \
&& sudo rpm -Uvh gdrcopy-2.5.2*.rpm \
&& sudo rpm -Uvh gdrcopy-devel-2.5.2*.rpm
```

Ubuntu 26.04, Ubuntu 24.04, and Ubuntu 22.04

Untuk menginstal GDRCopy

1. Instal dependensi yang diperlukan.

```
$ sudo apt-get install -y build-essential devscripts debhelper fakeroot pkg-
config dkms
```

> **Note**
>
> Di Ubuntu 22.04, instal juga dependensi tambahan berikut:

```
$ sudo apt-get install -y check libsubunit-dev
```

2. Unduh dan ekstrak paket GDRCopy, dan buat paketnya.

```
$ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.5.2.tar.gz \
&& tar xf v2.5.2.tar.gz \
&& cd gdrcopy-2.5.2/packages \
&& CUDA=/usr/local/cuda ./build-deb-packages.sh
```

3. Instal paket GDRCopy DEB.

```
$ sudo dpkg -i gdrdrv-dkms_2.5.2-1_amd64.*.deb \
&& sudo dpkg -i libgdrapi_2.5.2-1_amd64.*.deb \
&& sudo dpkg -i gdrcopy-tests_2.5.2-1_amd64.*.deb \
&& sudo dpkg -i gdrcopy_2.5.2-1_amd64.*.deb
```

### Debian 12

#### Untuk menginstal GDRCopy

1. Instal dependensi yang diperlukan.

```
$ sudo apt-get install -y build-essential devscripts debhelper fakeroot pkg-
config dkms
```

2. Mengunduh dan mengekstraksi paket GDRcopy.

```
$ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.5.2.tar.gz \
&& tar xf v2.5.2.tar.gz && cd gdrcopy-2.5.2/packages
```

3. Terapkan patch versi Debian dan buat paket.

```
$ sed -i 's/(2.5.2)/(2.5.2-1)/g' debian-lib/changelog \
&& sed -i 's/(2.5.2)/(2.5.2-1)/g' debian-tests/changelog \
&& sed -i 's/(2.5.2)/(2.5.2-1)/g' dkms/debian/changelog \
&& sed -i 's/(2.5.2)/(2.5.2-1)/g' debian-meta/changelog \
&& sed -i 's/FULL_VERSION="${VERSION}"/FULL_VERSION="${VERSION}-
${DEBIAN_VERSION}"/g' build-deb-packages.sh
```

```
$ CUDA=/usr/local/cuda ./build-deb-packages.sh
```

4. Instal paket GDRCopy DEB.

```
$ sudo dpkg -i gdrdrv-dkms_2.5.2*.deb \
&& sudo dpkg -i libgdrapi_2.5.2*.deb \
&& sudo dpkg -i gdrcopy-tests_2.5.2*.deb \
&& sudo dpkg -i gdrcopy_2.5.2*.deb
```

### RHEL 10

#### Untuk menginstal GDRCopy

1. Instal dependensi yang diperlukan.

```
$ sudo yum -y install dkms rpm-build make check check-devel
```

2. Mengunduh dan mengekstraksi paket GDRcopy.

```
$ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.5.2.tar.gz \
&& tar xf v2.5.2.tar.gz && cd gdrcopy-2.5.2/packages
```

3. Membangun paket RPM GDRCopy.

```
$ CUDA=/usr/local/cuda ./build-rpm-packages.sh
```

4. Instal paket RPM GDRCopy.

```
$ sudo rpm -Uvh gdrcopy-kmod-2.5.2*dkms*.rpm \
&& sudo rpm -Uvh gdrcopy-2.5.2*.rpm \
&& sudo rpm -Uvh gdrcopy-devel-2.5.2*.rpm
```

## Langkah 5: Instal perangkat lunak EFA

Instal EFA-enabled kernel, driver EFA, Libfabric, plugin aws-ofi-nccl, dan tumpukan Open MPI yang diperlukan untuk mendukung EFA pada instance Anda.

Untuk menginstal perangkat lunak EFA

1. Hubungkan ke instans yang Anda luncurkan. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..
2. Unduh file penginstalan perangkat lunak EFA. File instalasi perangkat lunak datang sebagai file tarball (`.tar.gz`) terkompresi. Untuk mengunduh versi stabil terbaru, gunakan perintah berikut.

   ```
   $ curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.50.0.tar.gz
   ```

   Anda juga bisa mendapatkan versi terbaru dengan mengganti nomor versi dengan `latest` dalam perintah sebelumnya.
3. (Opsional) Verifikasi keaslian dan integritas file tarball EFA (`.tar.gz`).

   Kami menyarankan Anda melakukan hal ini untuk memverifikasi identitas penerbit perangkat lunak dan memeriksa apakah file tersebut tidak diubah atau rusak sejak file tersebut diterbitkan. Jika Anda tidak ingin memverifikasi file tarball, lewati langkah ini.

> **Note**
>
> Atau, jika Anda memilih untuk memverifikasi file tarball menggunakan checksum MD5 atau SHA256, lihat Memverifikasi penginstal EFA menggunakan checksum.

a. Unduh kunci GPG publik dan impor ke dalam keyring Anda.

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer.key && gpg --
import aws-efa-installer.key
```

Perintah tersebut harus mengembalikan nilai kunci. Catat nilai kunci, karena Anda membutuhkannya di langkah selanjutnya.

b. Verifikasi sidik jari kunci GPG. Jalankan perintah berikut dan tentukan nilai kunci dari langkah sebelumnya.

```
$ gpg --fingerprint key_value
```

Perintah tersebut harus mengembalikan sidik jari yang identik dengan `4E90 91BC BB97 A96B 26B1 5E59 A054 80B1 DD2D 3CCC`. Jika sidik jari tidak cocok, jangan jalankan skrip instalasi EFA, dan hubungi Dukungan.

c. Unduh file tanda tangan dan verifikasi tanda tangan pada file tarball EFA.

```
$ wget https://efa-installer.amazonaws.com/aws-efa-installer-1.50.0.tar.gz.sig
 && gpg --verify ./aws-efa-installer-1.50.0.tar.gz.sig
```

Berikut ini adalah contoh output.

```
gpg: Signature made Wed 29 Jul 2020 12:50:13 AM UTC using RSA key ID DD2D3CCC
gpg: Good signature from "Amazon EC2 EFA <ec2-efa-maintainers@amazon.com>"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 4E90 91BC BB97 A96B 26B1  5E59 A054 80B1 DD2D 3CCC
```

Jika hasilnya mencakup `Good signature`, dan sidik jari cocok dengan sidik jari yang dikembalikan di langkah sebelumnya, lanjutkan ke langkah berikutnya. Jika tidak, jangan jalankan skrip instalasi EFA, dan hubungi Dukungan.

4. Ekstraksi file dari file `.tar.gz` yang dikompresi dan navigasi ke dalam direktori yang diekstraksi.

```
$ tar -xf aws-efa-installer-1.50.0.tar.gz && cd aws-efa-installer
```

5. (Opsional) Verifikasi tanda tangan paket individual selama instalasi.

   Dimulai dengan EFA installer 1.48.0, installer menyertakan paket RPM dan DEB GPG-signed individual. Untuk memverifikasi keaslian dan integritas masing-masing paket selama instalasi, gunakan ben `--check-signatures` dera. Saat Anda mengaktifkan tanda ini, penginstal memverifikasi semua tanda tangan paket terlebih dahulu, dan hanya melanjutkan penginstalan jika setiap paket lulus verifikasi. Jika ada paket yang gagal verifikasi, penginstal segera keluar tanpa menginstal apa pun.

   a. Unduh kunci publik GPG.

   ```
   $ wget https://efa-installer.amazonaws.com/aws-efa-installer.key
   ```

   b. Ekspor jalur kunci. Kemudian, pada langkah berikutnya, tambahkan `--check-signatures` ke perintah instalasi dan gunakan alih- `sudo -E` alih `sudo` untuk melestarikan variabel lingkungan.

   ```
   $ export EFA_INSTALLER_KEY=$(pwd)/aws-efa-installer.key
   ```

   Pada RPM-based sistem (Amazon Linux 2023, RHEL, Rocky Linux, dan SUSE), penginstal memverifikasi setiap RPM menggunakan. `rpm --checksig` Pada DEB-based sistem (Ubuntu, Debian), installer memverifikasi setiap DEB menggunakan verifikasi tanda tangan GPG. Jika verifikasi paket gagal, instalasi segera dibatalkan.

   > **Note**
   >
   > `--check-signatures`Bendera adalah opsional. Tanpa itu, penginstal tidak melakukan verifikasi tanda tangan individu.

6. Unduh skrip penginstalan perangkat lunak EFA.

> **Note**
>
> Jika Anda menyelesaikan langkah opsional sebelumnya untuk mengatur verifikasi tanda tangan paket, tambahkan `--check-signatures` ke perintah instalasi dan gunakan al `sudo -E` ih-alih`sudo`. Sebagai contoh: `sudo -E ./efa_installer.sh -y --mpi=openmpi5 --check-signatures`.

> **Note**
>
> Dari EFA 1.30.0, Open MPI 4.1 dan Open MPI 5 diinstal secara default. Kecuali Anda memerlukan Open MPI 4.1, instal hanya Open MPI 5. Perintah berikut menginstal Open MPI 5 saja. Jika Anda ingin menginstal Open MPI 4.1 dan Open MPI 5, hapus`--mpi=openmpi5`.

```
$ sudo ./efa_installer.sh -y --mpi=openmpi5
```

Libfabric diinstal di `/opt/amazon/efa` direktori. Plu gin aws-ofi-nccl diinstal di direktori. `/opt/amazon/ofi-nccl` Open MPI diinstal di `/opt/amazon/openmpi` direktori.

7. Jika penginstal EFA meminta Anda untuk me-reboot instans, lakukanlah dan kemudian sambungkan kembali ke instans. Jika tidak, log out dari instans lalu log kembali untuk menyelesaikan penginstalan.
8. Konfirmasikan bahwa perangkat lunak EFA berhasil diinstal.

```
$ fi_info -p efa -t FI_EP_RDM
```

Perintah tersebut harus mengembalikan informasi tentang antarmuka Libfabric EFA.

## Langkah 6: Instal NCCL

Instal NCCL. Untuk informasi lebih lanjut tentang NCCL, lihat repositori NCCL.

Untuk menginstal NCCL

1. Buka direktori `/opt` tersebut.

```
$ cd /opt
```

2. Gandakan repositori NCCL resmi ke instans dan navigasi ke repositori klon lokal.

```
$ sudo git clone https://github.com/NVIDIA/nccl.git -b v2.30.4-1 && cd nccl
```

3. Bangun dan instal NCCL dan tentukan direktori instalasi CUDA.

```
$ sudo make -j src.build CUDA_HOME=/usr/local/cuda-13
```

## Langkah 7: Instal uji NCCL

Instal uji NCCL. Uji NCCL memungkinkan Anda untuk mengonfirmasi bahwa NCCL diinstal dengan benar dan beroperasi sesuai harapan. Untuk informasi lebih lanjut tentang pengujian NCCL, lihat repositori nccl-tests.

Untuk menginstal pengujian EFA

1. Navigasi ke direktori beranda Anda.

```
$ cd $HOME
```

2. Gandakan repositori nccl-tests resmi ke instans dan navigasi ke repositori klon lokal.

```
$ git clone https://github.com/NVIDIA/nccl-tests.git && cd nccl-tests
```

3. Tambahkan direktori Libfabric ke variabel `LD_LIBRARY_PATH`.

   - Amazon Linux 2023

     ```
     $ export LD_LIBRARY_PATH=/opt/amazon/efa/lib64:$LD_LIBRARY_PATH
     ```

   - Ubuntu dan Debian

     ```
     $ export LD_LIBRARY_PATH=/opt/amazon/efa/lib:$LD_LIBRARY_PATH
     ```

   - HEEL 10

     ```
     $ export LD_LIBRARY_PATH=/opt/amazon/efa/lib64:$LD_LIBRARY_PATH
     ```

4. Instal pengujian NCCL dan tentukan direktori penginstalan MPI, NCCL, dan CUDA.

```
$ make MPI=1 MPI_HOME=/opt/amazon/openmpi NCCL_HOME=/opt/nccl/build CUDA_HOME=/usr/
local/cuda-13
```

## Langkah 8: Uji konfigurasi EFA dan NCCL Anda

Jalankan pengujian untuk memastikan bahwa instans sementara Anda sudah dikonfigurasi dengan benar untuk EFA dan NCCL.

Untuk menguji konfigurasi EFA dan NCCL Anda

1. Buat file host yang menentukan host untuk menjalankan pengujian. Perintah berikut membuat file host dengan nama `my-hosts` yang mencakup referensi ke instans itu sendiri.

   IMDSv2

   ```
   [ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H
    "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
   && curl -H "X-aws-ec2-metadata-token: $TOKEN" -v http://169.254.169.254/latest/
   meta-data/local-ipv4 >> my-hosts
   ```

   IMDSv1

   ```
   [ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/local-ipv4 >> my-
   hosts
   ```

2. Jalankan tes NCCL. Perintah berikut mengasumsikan bahwa Anda memiliki 8 GPU per instance. Sesuaikan `-N` nilai `-n` dan berdasarkan jenis instans Anda.

   ```
   $ /opt/amazon/openmpi/bin/mpirun \
   -x FI_EFA_USE_DEVICE_RDMA=1 \
   -x LD_LIBRARY_PATH=/opt/nccl/build/lib:/usr/local/cuda/lib64:/opt/amazon/efa/lib:/
   opt/amazon/openmpi/lib:/opt/amazon/ofi-nccl/lib:$LD_LIBRARY_PATH \
   -x NCCL_DEBUG=INFO \
   --hostfile my-hosts -n 8 -N 8 \
   --mca pml ^cm --mca btl tcp,self --mca btl_tcp_if_exclude lo,docker0 --bind-to none
    \
   $HOME/nccl-tests/build/all_reduce_perf -b 8 -e 1G -f 2 -g 1 -c 1 -n 100
   ```

3. Anda dapat mengonfirmasi bahwa EFA aktif sebagai penyedia pokok NCCL ketika log `NCCL_DEBUG` dicetak.

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Selected Provider is efa*
```

Informasi tambahan berikut ditampilkan saat menggunakan instans `p4d.24xlarge`.

```
ip-192-168-2-54:14:14 [0] NCCL INFO NET/OFI Running on P4d platform, Setting
 NCCL_TOPO_FILE environment variable to /home/ec2-user/install/plugin/share/aws-
ofi-nccl/xml/p4d-24xl-topo.xml
```

## Langkah 9: Instal aplikasi machine learning Anda

Instal aplikasi machine learning di instans sementara. Prosedur penginstalan bervariasi tergantung pada aplikasi machine learning tertentu. Untuk informasi selengkapnya tentang menginstal perangkat lunak pada instans Linux Anda, lihat Mengel ola pembaruan OS di Panduan Pengguna Amazon Linux 2023.

> **Note**
>
> Lihat dokumentasi aplikasi machine learning Anda untuk petunjuk penginstalan.

## Langkah 10: Buat EFA dan AMI NCCL-enabled

Setelah menginstal komponen perangkat lunak yang diperlukan, Anda membuat AMI yang dapat Anda gunakan kembali untuk meluncurkan EFA-enabled instans Anda.

Untuk membuat AMI dari instans sementara Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans sementara yang Anda buat dan pilih Tindakan, Gambar, Buat gambar.
4. Untuk Buat gambar, lakukan hal berikut:
   a. Untuk Nama gambar, masukkan nama deskriptif untuk AMI.
   b. (Opsional) Untuk Deskripsi gambar, masukkan deskripsi singkat tentang tujuan AMI.

c. Pilih Buat gambar.

5. Di panel navigasi, pilih AMI.
6. Temukan AMI yang Anda buat dalam daftar. Tunggu hingga status berubah dari `pending` menjadi `available` sebelum melanjutkan ke langkah berikutnya.

## Langkah 11: Mengakhiri instans sementara

Pada titik ini, Anda tidak memerlukan lagi instans sementara yang Anda luncurkan. Anda dapat mengakhiri instans agar biaya tidak dibebankan lagi padanya.

Untuk mengakhiri instans sementara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans sementara yang Anda buat dan pilih Tindakan, Status instans, Akhiri instans.
4. Saat diminta konfirmasi, pilih Akhiri.

## Langkah 12: Luncurkan EFA dan NCCL-enabled instance ke grup penempatan cluster

Luncurkan EFA dan NCCL-enabled instans Anda ke grup penempatan cluster menggunakan EFA-enabled AMI dan grup EFA-enabled keamanan yang Anda buat sebelumnya.

> **Note**
>
> - Ini bukan persyaratan mutlak untuk meluncurkan EFA-enabled instans Anda ke grup penempatan cluster. Namun, sebaiknya jalankan EFA-enabled instans Anda di grup penempatan cluster karena meluncurkan instans ke grup latensi rendah di Zona Ketersediaan tunggal.
> - Untuk memastikan kapasitas tersedia saat Anda menskalakan instans klaster, Anda dapat membuat Reservasi Kapasitas untuk grup penempatan klaster Anda. Untuk informasi selengkapnya, lihat Gunakan Reservasi Kapasitas dengan grup penempatan.

New console

Untuk meluncurkan instans sementara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih Luncurkan Instans untuk membuka wizard peluncuran instans baru.
3. (Opsional) Di bagian Nama dan tanda, berikan nama untuk instans, seperti `EFA-instance`. Nama ditetapkan ke instans sebagai tanda sumber daya (`Name=EFA-instance`).
4. Di bagian Application and OS Images, pilih AMI Saya, lalu pilih AMI yang Anda buat pada langkah sebelumnya.
5. Di bagian Tipe instans, pilih `p3dn.24xlarge` atau `p4d.24xlarge`.
6. Pada bagian Pasangan kunci, pilih pasangan kunci yang akan digunakan untuk instans.
7. Di bagian Pengaturan jaringan, pilih Edit, lalu lakukan hal berikut:
   a. Untuk Subnet, pilih subnet untuk meluncurkan instans. Jika Anda tidak memilih subnet, Anda tidak dapat mengaktifkan instans untuk EFA.
   b. Untuk Firewall (grup keamanan), pilih Pilih grup keamanan yang ada, lalu pilih grup keamanan yang Anda buat di langkah sebelumnya.
   c. Perluas bagian Konfigurasi jaringan lanjutan.

      Untuk Antarmuka jaringan 1, pilih Indeks kartu jaringan = 0, indeks perangkat = 0, dan tipe antarmuka = EFA dengan ENA.

      (Opsional) Jika Anda menggunakan jenis instance multi-kartu, seperti `p4d.24xlarge` ataup`5.48xlarge`, untuk setiap antarmuka jaringan tambahan yang diperlukan, pilih Tam bah antarmuka jaringan, untuk Indeks kartu jaringan pilih indeks yang tidak digunakan berikutnya, lalu pilih Indeks perangkat = 1 dan Jenis antarmuka = EFA dengan ENA atau. EFA-only
8. (Opsional) Di bagian Penyimpanan, konfigurasikan volume sesuai kebutuhan.
9. Di bagian Detail lanjutan, untuk nama grup Penempatan, pilih grup penempatan klaster untuk meluncurkan instans. Jika Anda perlu membuat grup penempatan klaster baru, pilih Buat grup penempatan baru.
10. Di panel Ringkasan di sebelah kanan, untuk Jumlah instance, masukkan jumlah EFA-enabled instance yang ingin Anda luncurkan, lalu pilih Luncurkan instance.

Old console

Untuk meluncurkan EFA dan NCCL-enabled instans Anda ke grup penempatan cluster

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Luncurkan Instans.
3. Di halaman Pilih AMI, pilih AMI saya, temukan AMI yang Anda buat sebelumnya, lalu pilih Pilih.
4. Di halaman Pilih Tipe Instans, pilih p3dn.24xlarge lalu pilih Berikutnya: Konfigurasikan Detail Instans.
5. Pada halaman Konfigurasi Detail Instans, lakukan langkah berikut:
   a. Untuk Jumlah instance, masukkan jumlah EFA dan NCCL-enabled instans yang ingin Anda luncurkan.
   b. Untuk Jaringan dan Subnet, pilih VPC dan subnet sebagai tujuan peluncuran instans.
   c. Untuk Grup penempatan, pilih Tambahkan instans ke grup penempatan.
   d. Untuk Nama grup penempatan, pilih Tambahkan ke grup penempatan baru, lalu masukkan nama deskriptif untuk grup penempatan. Lalu untuk Strategi grup penempatan, pilih klaster.
   e. Untuk EFA, pilih Aktifkan.
   f. Di bagian Antarmuka Jaringan, untuk perangkat eth0, pilih Antarmuka jaringan baru. Secara opsional, Anda dapat menentukan alamat IPv4 primer dan satu atau lebih alamat IPv4 sekunder. Jika Anda meluncurkan instans ke subnet yang memiliki blok CIDR IPv6 terkait, Anda dapat secara opsional menentukan alamat IPv6 primer dan satu atau lebih alamat IPv6 sekunder.
   g. Pilih Berikutnya: Tambahkan Penyimpanan.
6. Di halaman Tambahkan Penyimpanan, tentukan volume yang akan dipasang ke instans selain volume yang ditentukan oleh AMI (seperti volume perangkat root). Lalu, pilih Berikutnya: Tambahkan Tanda.
7. Di halaman Tambahkan Tanda, tentukan tanda untuk instans, seperti nama yang mudah digunakan, lalu pilih Selanjutnya: Konfigurasikan Grup Keamanan.
8. Di halaman Mengonfigurasi Grup Keamanan, untuk Tetapkan grup keamanan, pilih Pilih grup keamanan yang sudah ada, lalu pilih grup keamanan yang Anda buat sebelumnya.
9. Pilih Tinjau dan Luncurkan.

10. Di halaman Meninjau Peluncuran Instans, tinjau pengaturannya, lalu pilih Luncurkan untuk memilih key pair dan meluncurkan instans Anda.

## Langkah 13: Mengaktifkan SSH tanpa kata sandi

> **Note**
>
> Pengguna SSH default bervariasi menurut sistem operasi: `ubuntu` untuk Ubuntu, `admin` untuk Debian, `ec2-user` untuk Amazon Linux dan RHEL.

Agar aplikasi Anda dapat berjalan di semua instans dalam klaster, Anda harus mengaktifkan akses SSH tanpa kata sandi dari simpul pemimpin ke simpul anggota. Simpul pemimpin adalah instans dari mana Anda menjalankan aplikasi Anda. Instans yang tersisa di klaster adalah simpul anggota.

Untuk mengaktifkan SSH tanpa kata sandi antar instans dalam klaster

1. Pilih satu instans dalam klaster sebagai simpul pemimpin, dan hubungkan ke instans tersebut.
2. Nonaktifkan `strictHostKeyChecking` dan aktifkan `ForwardAgent` pada simpul pemimpin. Buka `~/.ssh/config` menggunakan editor teks pilihan Anda dan tambahkan berikut ini.

   ```
   Host *
       ForwardAgent yes
   Host *
       StrictHostKeyChecking no
   ```

3. Membuat pasangan kunci RSA.

   ```
   $ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
   ```

   Pasangan kunci dibuat dalam direktori `$HOME/.ssh/`.
4. Ubah izin kunci privat pada simpul pemimpin.

   ```
   $ chmod 600 ~/.ssh/id_rsa
   chmod 600 ~/.ssh/config
   ```

5. Buka `~/.ssh/id_rsa.pub` menggunakan editor teks pilihan Anda dan salin kunci.
6. Untuk setiap simpul anggota di klaster, lakukan hal berikut:

a. Hubungkan dengan instans.

b. Buka `~/.ssh/authorized_keys` menggunakan editor teks pilihan Anda tambahkan kunci publik yang Anda salin sebelumnya.

7. Untuk menguji apakah SSH tanpa kata sandi berfungsi seperti yang diharapkan, hubungkan ke simpul pemimpin Anda dan jalankan perintah berikut.

```
$ ssh member_node_private_ip
```

Anda harus terhubung ke simpul anggota tanpa diminta untuk memasukkan kunci atau kata sandi.

## Memulai dengan EFA dan NIXL untuk beban kerja inferensi di Amazon EC2

NVIDIA Inference Xfer Library (NIXL) adalah pustaka komunikasi throughput tinggi, latensi rendah yang dirancang khusus untuk beban kerja inferensi terpilah. NIXL dapat digunakan bersama dengan EFA dan Libfabric untuk mendukung KV-cache transfer antara node prefill dan decode, dan memungkinkan KV-cache pergerakan yang efisien antara berbagai lapisan penyimpanan. Untuk informasi lebih lanjut, lihat situs web NIXL.

Persyaratan

- Hanya AMI dasar Ubuntu 24.04 dan Ubuntu 22.04 yang didukung.
- EFA hanya mendukung NIXL 1.0.0 dan yang lebih baru.

Langkah 1: Siapkan grup EFA-enabled keamanan

EFA memerlukan grup keamanan yang memungkinkan semua lalu lintas masuk dan keluar ke dan dari grup keamanan itu sendiri. Prosedur berikut menciptakan grup keamanan yang memungkinkan semua lalu lintas masuk dan ke luar ke dan dari dirinya sendiri, dan yang memungkinkan lalu lintas SSH masuk dari alamat IPv4 apa pun untuk konektivitas SSH.

> **Important**
>
> Grup keamanan ini ditujukan untuk tujuan pengujian saja. Untuk lingkungan produksi Anda, kami sarankan Anda membuat aturan SSH masuk yang memungkinkan lalu lintas hanya dari

alamat IP dari mana Anda terhubung, seperti alamat IP komputer Anda, atau berbagai alamat IP di jaringan lokal Anda.

Untuk skenario lainnya, lihat Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda.

Untuk membuat grup EFA-enabled keamanan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dalam panel navigasi, pilih Grup Keamanan, lalu pilih Buat grup keamanan.
3. Di jendela Buat Grup Keamanan, lakukan hal berikut:
   a. Untuk Nama grup keamanan, masukkan nama deskriptif untuk grup keamanan, seperti `EFA-enabled security group`.
   b. (Opsional) Untuk Deskripsi, masukkan deskripsi singkat grup keamanan.
   c. Untuk VPC, pilih VPC tempat Anda ingin meluncurkan instans Anda EFA-enabled .
   d. Pilih Buat grup keamanan.
4. Pilih grup keamanan yang Anda buat, dan pada tab Detail, salin ID grup keamanan.
5. Dengan grup keamanan yang masih dipilih, pilih Tindakan, Edit aturan masuk, lalu lakukan hal berikut:
   a. Pilih Tambahkan aturan.
   b. Untuk Tipe, pilih Semua lalu lintas.
   c. Untuk Tipe sumber, pilih Kustom dan tempelkan ID grup keamanan yang Anda salin ke dalam bidang.
   d. Pilih Tambahkan aturan.
   e. Untuk Tipe, pilih SSH.
   f. Untuk Jenis Sumber, pilih Anywhere-IPv4.
   g. Pilih Simpan aturan.
6. Dengan grup keamanan yang masih dipilih, pilih Tindakan, Edit aturan keluar, lalu lakukan hal berikut:
   a. Pilih Tambahkan aturan.
   b. Untuk Tipe, pilih Semua lalu lintas.

c. Untuk Tipe tujuan, pilih Kustom dan tempelkan ID grup keamanan yang Anda salin ke dalam bidang.

d. Pilih Simpan aturan.

> **Important**
>
> Aturan masuk dan keluar yang merujuk sendiri (memungkinkan semua lalu lintas ke dan dari grup keamanan itu sendiri) wajib bagi EFA untuk berfungsi. Tanpa aturan ini, lalu lintas EFA antar instance akan diblokir.

### Langkah 2: Luncurkan instans sementara

Luncurkan instans sementara yang dapat Anda gunakan untuk menginstal dan mengonfigurasi komponen perangkat lunak EFA. Anda menggunakan instance ini untuk membuat EFA-enabled AMI dari mana Anda dapat meluncurkan EFA-enabled instans Anda.

Untuk meluncurkan instans sementara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih Luncurkan Instans untuk membuka wizard peluncuran instans baru.
3. (Opsional) Di bagian Nama dan tanda, berikan nama untuk instans, seperti `EFA-instance`. Nama ditetapkan ke instans sebagai tanda sumber daya (`Name=EFA-instance`).
4. Di bagian Aplikasi dan Gambar OS, pilih AMI untuk salah satu sistem operasi yang didukung. Anda juga dapat memilih DLAMI yang didukung yang ditemukan di Halaman Catatan Rilis DLAMI.
5. Di bagian Jenis Instance, pilih jenis instance yang didukung.
6. Pada bagian Pasangan kunci, pilih pasangan kunci yang akan digunakan untuk instans.
7. Di bagian Pengaturan jaringan, pilih Edit, lalu lakukan hal berikut:

   a. Untuk Subnet, pilih subnet untuk meluncurkan instans. Jika Anda tidak memilih subnet, Anda tidak dapat mengaktifkan instans untuk EFA.

   b. Untuk Firewall (grup keamanan), pilih Pilih grup keamanan yang ada, lalu pilih grup keamanan yang Anda buat di langkah sebelumnya.

   c. Perluas bagian Konfigurasi jaringan lanjutan.

Untuk Antarmuka jaringan 1, pilih Indeks kartu jaringan = 0, indeks perangkat = 0, dan tipe antarmuka = EFA dengan ENA.

(Opsional) Jika Anda menggunakan jenis instance multi-kartu, seperti `p4d.24xlarge` ataup`5.48xlarge`, untuk setiap antarmuka jaringan tambahan yang diperlukan, pilih Tam bah antarmuka jaringan, untuk Indeks kartu jaringan pilih indeks yang tidak digunakan berikutnya, lalu pilih Indeks perangkat = 1 dan Jenis antarmuka = EFA dengan ENA atau. EFA-only

8. Di bagian Penyimpanan, konfigurasikan volume sesuai kebutuhan.

> **Note**
>
> Anda harus menyediakan penyimpanan tambahan 10 hingga 20 GiB untuk Nvidia CUDA. Jika Anda tidak menyediakan penyimpanan yang cukup, Anda akan menerima kesalahan `insufficient disk space` saat mencoba menginstal driver Nvidia dan toolkit CUDA.

9. Di panel Ringkasan di sebelah kanan, pilih Luncurkan instans.

> **Important**
>
> Lewati Langkah 3 jika AMI Anda sudah menyertakan driver GPU Nvidia, toolkit CUDA, dan cuDNN, atau jika Anda menggunakan instance non-GPU.

Langkah 3: Instal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN

Untuk menginstal driver Nvidia GPU, kit alat CUDA Nvidia, dan cuDNN

1. Untuk memastikan bahwa semua paket perangkat lunak Anda telah diperbarui, lakukan pembaruan perangkat lunak cepat di instans Anda.

```
$ sudo apt-get update && sudo apt-get upgrade -y
```

2. Pasang utilitas yang diperlukan untuk memasang driver GPU Nvidia dan toolkit Nvidia CUDA.

```
$ sudo apt-get install build-essential -y
```

3. Untuk menggunakan driver GPU Nvidia, Anda harus menonaktifkan driver sumber terbuka `nouveau` terlebih dahulu.

   a. Instal utilitas yang diperlukan dan paket header kernel untuk versi kernel yang sedang Anda jalankan.

   ```
   $ sudo apt-get install -y gcc make linux-headers-$(uname -r)
   ```

   b. Tambahkan `nouveau` ke file daftar penolakan `/etc/modprobe.d/blacklist.conf` .

   ```
   $ cat << EOF | sudo tee --append /etc/modprobe.d/blacklist.conf
   blacklist vga16fb
   blacklist nouveau
   blacklist rivafb
   blacklist nvidiafb
   blacklist rivatv
   EOF
   ```

   c. Buka `/etc/default/grub` menggunakan editor teks pilihan Anda dan tambahkan berikut ini.

   ```
   GRUB_CMDLINE_LINUX="rdblacklist=nouveau"
   ```

   d. Buat kembali konfigurasi Grub.

   ```
   $ sudo update-grub
   ```

4. Boot ulang dan terhubung kembali ke instans Anda.

5. Tambahkan repositori CUDA dan instal driver Nvidia GPU, toolkit CUDA NVIDIA, dan cuDNN.

```
$ sudo apt-key adv --fetch-keys http://developer.download.nvidia.com/compute/
machine-learning/repos/ubuntu2004/x86_64/7fa2af80.pub \
&& wget -O /tmp/deeplearning.deb http://developer.download.nvidia.com/
compute/machine-learning/repos/ubuntu2004/x86_64/nvidia-machine-learning-repo-
ubuntu2004_1.0.0-1_amd64.deb \
&& sudo dpkg -i /tmp/deeplearning.deb \
&& wget -O /tmp/cuda.pin https://developer.download.nvidia.com/compute/cuda/repos/
ubuntu2004/x86_64/cuda-ubuntu2004.pin \
&& sudo mv /tmp/cuda.pin /etc/apt/preferences.d/cuda-repository-pin-600 \
&& sudo apt-key adv --fetch-keys https://developer.download.nvidia.com/compute/
cuda/repos/ubuntu2004/x86_64/3bf863cc.pub \
```

```
&& sudo add-apt-repository 'deb http://developer.download.nvidia.com/compute/cuda/
repos/ubuntu2004/x86_64/ /' \
&& sudo apt update \
&& sudo apt install nvidia-dkms-535 \
&& sudo apt install -o Dpkg::Options::='--force-overwrite' cuda-drivers-535 cuda-
toolkit-12-3 libcudnn8 libcudnn8-dev -y
```

6. Boot ulang dan terhubung kembali ke instans Anda.
7. (`p4d.24xlarge` dan `p5.48xlarge` saja) Instal Nvidia Fabric Manager.

   a. Anda harus menginstal versi Nvidia Fabric Manager yang cocok dengan versi modul kernel Nvidia yang Anda instal pada langkah sebelumnya.

      Jalankan perintah berikut untuk menentukan versi modul kernel Nvidia.

      ```
      $ cat /proc/driver/nvidia/version | grep "Kernel Module"
      ```

      Berikut ini adalah output contoh.

      ```
      NVRM version: NVIDIA UNIX x86_64 Kernel Module  450.42.01  Tue Jun 15 21:26:37
       UTC 2021
      ```

      Pada contoh di atas, versi utama `450` dari modul kernel diinstal. Ini berarti Anda perlu menginstal versi Nvidia Fabric Manager `450`.

   b. Instal Nvidia Fabric Manager. Jalankan perintah berikut dan tentukan versi utama yang diidentifikasi pada langkah sebelumnya.

      ```
      $ sudo apt install -o Dpkg::Options::='--force-overwrite' nvidia-
      fabricmanager-major_version_number
      ```

      Misalnya, jika versi `450` utama modul kernel diinstal, gunakan perintah berikut untuk menginstal versi Nvidia Fabric Manager yang cocok.

      ```
      $ sudo apt install -o Dpkg::Options::='--force-overwrite' nvidia-
      fabricmanager-450
      ```

   c. Mulai layanan, dan pastikan bahwa layanan tersebut dimulai secara otomatis ketika instans dimulai. Nvidia Fabric Manager diperlukan untuk NV Switch Management.

```
$ sudo systemctl start nvidia-fabricmanager && sudo systemctl enable nvidia-
fabricmanager
```

8. Pastikan bahwa jalur CUDA diatur setiap kali instans dimulai.

   - Untuk shell bash, tambahkan pernyataan berikut ke /home/*username*/.bashrc dan /home/*username*/.bash_profile.

     ```
     export PATH=/usr/local/cuda/bin:$PATH
     export LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/lib64:
     $LD_LIBRARY_PATH
     ```

   - Untuk shell tcsh, tambahkan pernyataan berikut ke /home/*username*/.cshrc.

     ```
     setenv PATH=/usr/local/cuda/bin:$PATH
     setenv LD_LIBRARY_PATH=/usr/local/cuda/lib64:/usr/local/cuda/extras/CUPTI/lib64:
     $LD_LIBRARY_PATH
     ```

9. Untuk mengonfirmasi bahwa driver Nvidia GPU berfungsi, jalankan perintah berikut.

   ```
   $ nvidia-smi -q | head
   ```

   Perintah tersebut harus mengembalikan informasi tentang Nvidia GPU, driver Nvidia GPU, dan kit alat Nvidia CUDA.

> **Important**
>
> Lewati Langkah 4 jika AMI Anda sudah menyertakan GDRCopy, atau jika Anda menggunakan instance non-GPU.

### Langkah 4: Instal GDRCopy

Instal GDRCopy untuk meningkatkan kinerja Libfabric pada platform. GPU-based Untuk informasi selengkapnya tentang GDRCopy, lihat repositori GDRCopy.

Untuk menginstal GDRCopy

1. Instal dependensi yang diperlukan.

```
$ sudo apt -y install build-essential devscripts debhelper check libsubunit-dev
 fakeroot pkg-config dkms
```

2. Mengunduh dan mengekstraksi paket GDRcopy.

```
$ wget https://github.com/NVIDIA/gdrcopy/archive/refs/tags/v2.4.tar.gz \
&& tar xf v2.4.tar.gz \
&& cd gdrcopy-2.4/packages
```

3. Membangun paket GDRCopy DEB.

```
$ CUDA=/usr/local/cuda ./build-deb-packages.sh
```

4. Instal paket GDRCopy DEB.

```
$ sudo dpkg -i gdrdrv-dkms_2.4-1_amd64.*.deb \
&& sudo dpkg -i libgdrapi_2.4-1_amd64.*.deb \
&& sudo dpkg -i gdrcopy-tests_2.4-1_amd64.*.deb \
&& sudo dpkg -i gdrcopy_2.4-1_amd64.*.deb
```

> **Important**
>
> Lewati Langkah 5 jika AMI Anda sudah menyertakan penginstal EFA terbaru.

### Langkah 5: Instal perangkat lunak EFA

Instal EFA-enabled kernel, driver EFA, dan tumpukan Libfabric yang diperlukan untuk mendukung EFA pada instans Anda.

Untuk menginstal perangkat lunak EFA

1. Hubungkan ke instans yang Anda luncurkan. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..
2. Unduh file penginstalan perangkat lunak EFA. File penginstalan perangkat lunak dikemas menjadi file tarball (`.tar.gz`) yang dikompresi. Untuk mengunduh versi stabil terbaru, gunakan perintah berikut.

```
$ curl -O https://efa-installer.amazonaws.com/aws-efa-installer-1.50.0.tar.gz
```

3. Ekstrak file dari `.tar.gz` file terkompresi, hapus tarball, dan navigasikan ke direktori yang diekstrak.

```
$ tar -xf aws-efa-installer-1.50.0.tar.gz && rm -rf aws-efa-installer-1.50.0.tar.gz
 && cd aws-efa-installer
```

4. (Opsional) Verifikasi tanda tangan paket individual selama penginstalan.

   Dimulai dengan EFA installer 1.48.0, installer menyertakan paket RPM dan DEB GPG-signed individual. Untuk memverifikasi keaslian dan integritas masing-masing paket selama instalasi, gunakan ben `--check-signatures` dera. Saat Anda mengaktifkan tanda ini, penginstal memverifikasi semua tanda tangan paket terlebih dahulu, dan hanya melanjutkan penginstalan jika setiap paket lulus verifikasi. Jika ada paket yang gagal verifikasi, penginstal segera keluar tanpa menginstal apa pun.

   a. Unduh kunci publik GPG.

   ```
   $ wget https://efa-installer.amazonaws.com/aws-efa-installer.key
   ```

   b. Ekspor jalur kunci. Kemudian, pada langkah berikutnya, tambahkan `--check-signatures` ke perintah instalasi dan gunakan alih- `sudo -E` alih `sudo` untuk melestarikan variabel lingkungan.

   ```
   $ export EFA_INSTALLER_KEY=$(pwd)/aws-efa-installer.key
   ```

   Pada RPM-based sistem (Amazon Linux, RHEL, Rocky Linux, dan SUSE), penginstal memverifikasi setiap RPM menggunakan. `rpm --checksig` Pada DEB-based sistem (Ubuntu, Debian), installer memverifikasi setiap DEB menggunakan verifikasi tanda tangan GPG.

   Jika verifikasi paket apa pun gagal, instalasi segera dibatalkan, melindungi sistem Anda dari paket yang rusak atau berbahaya.

> **Note**
>
> `--check-signatures`Bendera adalah opsional. Tanpa itu, penginstal tidak melakukan verifikasi tanda tangan individu.

5. Unduh skrip penginstalan perangkat lunak EFA.

> **Note**
>
> Jika Anda menyelesaikan langkah opsional sebelumnya untuk mengatur verifikasi tanda tangan paket, tambahkan `--check-signatures` ke perintah instalasi dan gunakan al `sudo -E` ih-alihsudo. Sebagai contoh: `sudo -E ./efa_installer.sh -y --check-signatures`.

```
$ sudo ./efa_installer.sh -y
```

Libfabric diinstal di `/opt/amazon/efa` direktori.

6. Jika penginstal EFA meminta Anda untuk me-reboot instans, lakukanlah dan kemudian sambungkan kembali ke instans. Jika tidak, log out dari instans lalu log kembali untuk menyelesaikan penginstalan.
7. Pastikan bahwa komponen perangkat lunak EFA telah berhasil diinstal.

```
$ fi_info -p efa -t FI_EP_RDM
```

Perintah tersebut harus mengembalikan informasi tentang antarmuka Libfabric EFA. Contoh berikut menunjukkan output perintah.

- `p3dn.24xlarge` dengan satu antarmuka jaringan

```
provider: efa
fabric: EFA-fe80::94:3dff:fe89:1b70
domain: efa_0-rdm
version: 2.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
```

- `p4d.24xlarge` dan `p5.48xlarge` dengan berbagai antarmuka jaringan

```
provider: efa
fabric: EFA-fe80::c6e:8fff:fef6:e7ff
domain: efa_0-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::c34:3eff:feb2:3c35
domain: efa_1-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::c0f:7bff:fe68:a775
domain: efa_2-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
provider: efa
fabric: EFA-fe80::ca7:b0ff:fea6:5e99
domain: efa_3-rdm
version: 111.0
type: FI_EP_RDM
protocol: FI_PROTO_EFA
```

Langkah 6: Instal NIXL

Instal NIXL. Untuk informasi selengkapnya tentang NIXL, lihat repositori NIXL.

Pre-built distributions

Untuk menginstal NIXL menggunakan PyPI

1. Instal dependensi yang diperlukan.

```
$ sudo apt install pip
```

2. Instal NIXL.

```
$ pip install nixl
```

Build from source

Untuk membangun dan menginstal NIXL dari sumber

1. Instal dependensi yang diperlukan.

```
$ sudo apt install cmake pkg-config meson pybind11-dev libaio-dev nvidia-cuda-
toolkit pip libhwloc-dev \
&& pip install meson ninja pybind11
```

2. Navigasi ke direktori beranda Anda.

```
$ cd $HOME
```

3. Kloning repositori NIXL resmi ke instance dan navigasikan ke repositori kloning lokal.

```
$ sudo git clone https://github.com/ai-dynamo/nixl.git && cd nixl
```

4. Bangun dan instal NIXL dan tentukan jalur ke direktori instalasi Libfabric.

```
$ sudo meson setup . nixl --prefix=/usr/local/nixl -Dlibfabric_path=/opt/amazon/
efa
$ cd nixl && sudo ninja && sudo ninja install
```

Langkah 7: Instal NIXL Benchmark dan uji konfigurasi EFA dan NIXL Anda

Instal NIXL Benchmark dan jalankan pengujian untuk memastikan bahwa instance sementara Anda dikonfigurasi dengan benar untuk EFA dan NIXL. Benchmark NIXL memungkinkan Anda untuk mengonfirmasi bahwa NIXL diinstal dengan benar dan beroperasi seperti yang diharapkan. Untuk informasi selengkapnya, lihat repositori nixlbench.

NIXL Benchmark (nixlbench) membutuhkan ETCD untuk koordinasi antara klien dan server. Untuk menggunakan ETCD dengan NIXL memerlukan Server dan Klien ETCD, dan ETCD CPP API.

### Build from Docker

Untuk menginstal dan menguji NIXL Benchmark menggunakan Docker

1. Kloning repositori NIXL resmi ke instance dan arahkan ke direktori build nixlbench.

```
$ git clone https://github.com/ai-dynamo/nixl.git
$ cd nixl/benchmark/nixlbench/contrib
```

2. Bangun wadah.

```
$ ./build.sh
```

Untuk informasi selengkapnya tentang opsi build Docker, lihat repositori nixlbench.

3. Pasang Docker.

```
$ sudo apt install docker.io -y
```

4. Mulai server ETCD untuk koordinasi.

```
$ docker run -d --name etcd-server \
    -p 2379:2379 -p 2380:2380 \
    quay.io/coreos/etcd:v3.5.18 \
    /usr/local/bin/etcd \
    --data-dir=/etcd-data \
    --listen-client-urls=http://0.0.0.0:2379 \
    --advertise-client-urls=http://0.0.0.0:2379 \
    --listen-peer-urls=http://0.0.0.0:2380 \
    --initial-advertise-peer-urls=http://0.0.0.0:2380 \
    --initial-cluster=default=http://0.0.0.0:2380
```

5. Memvalidasi bahwa server ETCD sedang berjalan.

```
$ curl -L http://localhost:2379/health
```

Keluaran yang diharapkan

```
{"health":"true"}
```

6. Buka dua terminal untuk contoh. Pada kedua terminal, jalankan perintah berikut untuk memverifikasi instalasi. Perintah menggunakan server ETCD pada instance yang sama, menggunakan Libfabric sebagai backend, dan beroperasi menggunakan memori GPU.

```
$ docker run -it --gpus all --network host nixlbench:latest \
    nixlbench --etcd_endpoints http://localhost:2379 \
    --backend LIBFABRIC \
    --initiator_seg_type VRAM \
    --target_seg_type VRAM
```

> **Note**
>
> Gunakan nilai al `DRAM` ih-alih `VRAM` untuk instans non-GPU.

Build from source

> **Important**
>
> Ikuti tab ini hanya jika Anda memilih  Bangun dari sumber  di Langkah 6.

Untuk menginstal NIXL Benchmark

1. Instal dependensi sistem yang diperlukan.

```
$ sudo apt install libgflags-dev
```

2. Instal Server dan Klien ETCD.

```
$ sudo apt install -y etcd-server etcd-client
```

3. Instal ETCD CPP API.

   a. Instal dependensi yang diperlukan untuk ETCD CPP API.

```
$ sudo apt install libboost-all-dev libssl-dev libgrpc-dev libgrpc++-dev
 libprotobuf-dev protobuf-compiler-grpc libcpprest-dev
```

   b. Kloning dan instal ETCD CPP API.

```
$ cd $HOME
$ git clone https://github.com/etcd-cpp-apiv3/etcd-cpp-apiv3.git
$ cd etcd-cpp-apiv3
$ mkdir build && cd build
$ cmake ..
$ sudo make -j$(nproc) && sudo make install
```

4. Membangun dan menginstal nixlbench.

```
$ sudo meson setup . $HOME/nixl/benchmark/nixlbench -Dnixl_path=/usr/local/nixl/
$ sudo ninja && sudo ninja install
```

Untuk menguji konfigurasi EFA dan NIXL Anda

1. Mulai server ETCD pada instance.

```
$ etcd --listen-client-urls "http://0.0.0.0:2379" \
    --advertise-client-urls "http://localhost:2379" &
```

2. Memvalidasi bahwa server ETCD sedang berjalan.

```
$ curl -L http://localhost:2379/health
```

Keluaran yang diharapkan

```
{"health":"true"}
```

3. Buka dua terminal untuk contoh. Pada kedua terminal, selesaikan langkah-langkah berikut untuk menjalankan nixlbench.

   a. Arahkan ke direktori tempat nixlbench diinstal.

   ```
   $ cd /usr/local/nixlbench/bin/
   ```

   b. Jalankan tes dan tentukan backend, alamat server ETCD, dan jenis segmen inisiator. Perintah berikut menggunakan server ETCD pada instance yang sama, menggunakan Libfabric sebagai backend, dan beroperasi menggunakan memori GPU. Variabel lingkungan mengkonfigurasi yang berikut:

- `NIXL_LOG_LEVEL=INFO`- Mengaktifkan keluaran debugging terperinci. Anda juga dapat menentukan `WARN` untuk menerima hanya pesan kesalahan.
- `LD_LIBRARY_PATH`— Menetapkan jalur untuk perpustakaan NIXL.

Untuk informasi lebih lanjut tentang argumen Benchmark NIXL, lihat NIXLBench README di repositori nixlbench resmi.

```
$ export NIXL_LOG_LEVEL=INFO
$ export LD_LIBRARY_PATH=/usr/local/nixl/lib/$(gcc -dumpmachine):
$LD_LIBRARY_PATH

$ nixlbench --etcd-endpoints 'http://localhost:2379' \
    --backend 'LIBFABRIC' \
    --initiator_seg_type 'VRAM' \
    --target_seg_type 'VRAM'
```

> **Note**
>
> Gunakan nilai al `DRAM` ih-alih `VRAM` untuk instans non-GPU.

Langkah 8: Instal aplikasi pembelajaran mesin Anda

Instal aplikasi machine learning di instans sementara. Prosedur penginstalan bervariasi tergantung pada aplikasi machine learning tertentu.

> **Note**
>
> Lihat dokumentasi aplikasi pembelajaran mesin Anda untuk petunjuk penginstalan.

Langkah 9: Buat EFA dan AMI NIXL-enabled

Setelah menginstal komponen perangkat lunak yang diperlukan, Anda membuat AMI yang dapat Anda gunakan kembali untuk meluncurkan EFA-enabled instans Anda.

Untuk membuat AMI dari instans sementara Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih instans sementara yang Anda buat dan pilih Tindakan, Gambar, Buat gambar.
4. Untuk Buat gambar, lakukan hal berikut:
   a. Untuk Nama gambar, masukkan nama deskriptif untuk AMI.
   b. (Opsional) Untuk Deskripsi gambar, masukkan deskripsi singkat tentang tujuan AMI.
   c. Pilih Buat gambar.
5. Di panel navigasi, pilih AMI.
6. Temukan AMI yang Anda buat dalam daftar. Tunggu hingga status berubah dari `pending` menjadi `available` sebelum melanjutkan ke langkah berikutnya.

### Langkah 10: Mengakhiri instans sementara

Pada titik ini, Anda tidak memerlukan lagi instans sementara yang Anda luncurkan. Anda dapat mengakhiri instans agar biaya tidak dibebankan lagi padanya.

Untuk mengakhiri instans sementara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans sementara yang Anda buat dan pilih Tindakan, Status instans, Akhiri instans.
4. Saat diminta konfirmasi, pilih Akhiri.

### Langkah 11: Luncurkan EFA dan instans NIXL-enabled

Luncurkan EFA dan NIXL-enabled instans menggunakan EFA-enabled AMI yang Anda buat di Langkah 9, dan grup EFA-enabled keamanan yang Anda buat di Langkah 1.

Untuk meluncurkan EFA dan instans NIXL-enabled

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, lalu pilih Luncurkan Instans untuk membuka wizard peluncuran instans baru.
3. (Opsional) Di bagian Nama dan tanda, berikan nama untuk instans, seperti `EFA-instance`. Nama ditetapkan ke instans sebagai tanda sumber daya (`Name=EFA-instance`).

4. Di bagian Application and OS Images, pilih AMI Saya, lalu pilih AMI yang Anda buat pada langkah sebelumnya.
5. Di bagian Jenis Instance, pilih jenis instance yang didukung.
6. Pada bagian Pasangan kunci, pilih pasangan kunci yang akan digunakan untuk instans.
7. Di bagian Pengaturan jaringan, pilih Edit, lalu lakukan hal berikut:

   a. Untuk Subnet, pilih subnet untuk meluncurkan instans. Jika Anda tidak memilih subnet, Anda tidak dapat mengaktifkan instans untuk EFA.

   b. Untuk Firewall (grup keamanan), pilih Pilih grup keamanan yang ada, lalu pilih grup keamanan yang Anda buat di Langkah 1.

   c. Perluas bagian Konfigurasi jaringan lanjutan.

      Untuk Antarmuka jaringan 1, pilih Indeks kartu jaringan = 0, indeks perangkat = 0, dan tipe antarmuka = EFA dengan ENA.

      (Opsional) Jika Anda menggunakan jenis instance multi-kartu, seperti `p4d.24xlarge` ataup`5.48xlarge`, untuk setiap antarmuka jaringan tambahan yang diperlukan, pilih Tam bah antarmuka jaringan, untuk Indeks kartu jaringan pilih indeks yang tidak digunakan berikutnya, lalu pilih Indeks perangkat = 1 dan Jenis antarmuka = EFA dengan ENA atau. EFA-only
8. (Opsional) Di bagian Penyimpanan, konfigurasikan volume sesuai kebutuhan.
9. Di panel Ringkasan di sebelah kanan, untuk Jumlah instance, masukkan jumlah EFA-enabled instance yang ingin Anda luncurkan, lalu pilih Luncurkan instance.

### Langkah 12: Mengaktifkan SSH tanpa kata sandi

Agar aplikasi Anda dapat berjalan di semua instans dalam klaster, Anda harus mengaktifkan akses SSH tanpa kata sandi dari simpul pemimpin ke simpul anggota. Simpul pemimpin adalah instans dari mana Anda menjalankan aplikasi Anda. Instans yang tersisa di klaster adalah simpul anggota.

Untuk mengaktifkan SSH tanpa kata sandi antar instans dalam klaster

1. Pilih satu instans dalam klaster sebagai simpul pemimpin, dan hubungkan ke instans tersebut.
2. Nonaktifkan `strictHostKeyChecking` dan aktifkan `ForwardAgent` pada simpul pemimpin. Buka `~/.ssh/config` menggunakan editor teks pilihan Anda dan tambahkan berikut ini.

```
Host *
```

```
    ForwardAgent yes
Host *
    StrictHostKeyChecking no
```

3. Membuat pasangan kunci RSA.

```
$ ssh-keygen -t rsa -N "" -f ~/.ssh/id_rsa
```

Pasangan kunci dibuat dalam direktori `$HOME/.ssh/`.

4. Ubah izin kunci privat pada simpul pemimpin.

```
$ chmod 600 ~/.ssh/id_rsa
chmod 600 ~/.ssh/config
```

5. Buka `~/.ssh/id_rsa.pub` menggunakan editor teks pilihan Anda dan salin kunci.
6. Untuk setiap simpul anggota di klaster, lakukan hal berikut:
   a. Hubungkan dengan instans.
   b. Buka `~/.ssh/authorized_keys` menggunakan editor teks pilihan Anda tambahkan kunci publik yang Anda salin sebelumnya.
7. Untuk menguji apakah SSH tanpa kata sandi berfungsi seperti yang diharapkan, hubungkan ke simpul pemimpin Anda dan jalankan perintah berikut.

```
$ ssh member_node_private_ip
```

Anda harus terhubung ke simpul anggota tanpa diminta untuk memasukkan kunci atau kata sandi.

> **Important**
>
> Ikuti Langkah 13 hanya jika Anda mengikuti Langkah 7.

Langkah 13: Menguji konfigurasi EFA dan NIXL di seluruh instans

Jalankan pengujian untuk memastikan instans Anda dikonfigurasi dengan benar untuk EFA dan NIXL.

### Build from Docker

Untuk menguji konfigurasi EFA dan NIXL Anda di seluruh instans menggunakan Docker

1. Pilih dua host untuk menjalankan benchmark nixlbench. Gunakan alamat IP host pertama sebagai IP server ETCD untuk pertukaran metadata.
2. Mulai server ETCD di host 1.

```
$ docker run -d --name etcd-server \
    -p 2379:2379 -p 2380:2380 \
    quay.io/coreos/etcd:v3.5.18 \
    /usr/local/bin/etcd \
    --data-dir=/etcd-data \
    --listen-client-urls=http://0.0.0.0:2379 \
    --advertise-client-urls=http://0.0.0.0:2379 \
    --listen-peer-urls=http://0.0.0.0:2380 \
    --initial-advertise-peer-urls=http://0.0.0.0:2380 \
    --initial-cluster=default=http://0.0.0.0:2380
```

3. Memvalidasi bahwa server ETCD sedang berjalan.

```
$ curl -L http://localhost:2379/health
```

```
{"health":"true"}
```

4. Jalankan benchmark nixlbench pada host 1.

```
$ docker run -it --gpus all --network host nixlbench:latest \
    nixlbench --etcd_endpoints http://localhost:2379 \
    --backend LIBFABRIC \
    --initiator_seg_type VRAM
```

5. Jalankan benchmark nixlbench di host 2.

```
$ docker run -it --gpus all --network host nixlbench:latest \
    nixlbench --etcd_endpoints http://ETCD_SERVER_IP:2379 \
    --backend LIBFABRIC \
    --initiator_seg_type VRAM
```

Build from source

> **Important**
>
> Ikuti tab ini hanya jika Anda memilih Bangun dari sumber di Langkah 6.

Untuk menguji konfigurasi EFA dan NIXL di seluruh instans

1. Pilih dua host untuk menjalankan benchmark nixlbench. Gunakan alamat IP host pertama sebagai IP server ETCD untuk pertukaran metadata.
2. Luncurkan server ETCD di host 1.

```
$ etcd --listen-client-urls "http://0.0.0.0:2379" \
    --advertise-client-urls "http://localhost:2379" &
```

3. Memvalidasi bahwa server ETCD sedang berjalan.

```
$ curl -L http://localhost:2379/health
```

```
{"health":"true"}
```

4. Jalankan benchmark nixlbench pada host 1.

```
$ export NIXL_LOG_LEVEL=INFO
$ export LD_LIBRARY_PATH=$HOME/nixl/lib/x86_64-linux-gnu:$LD_LIBRARY_PATH

$ nixlbench \
    --etcd-endpoints http://localhost:2379 \
    --backend LIBFABRIC \
    --initiator_seg_type VRAM
```

5. Jalankan benchmark nixlbench di host 2.

```
$ export NIXL_LOG_LEVEL=INFO
$ export LD_LIBRARY_PATH=$HOME/nixl/lib/x86_64-linux-gnu:$LD_LIBRARY_PATH

$ nixlbench \
    --etcd-endpoints http://ETCD_SERVER_IP:2379 \
    --backend LIBFABRIC \
```

```
    --initiator_seg_type VRAM
```

Langkah 14: Uji penyajian inferensi terpilah melalui vLLM (Opsional)

Setelah NIXL diinstal, Anda dapat menggunakan NIXL melalui inferensi LLM dan kerangka kerja penyajian seperti vLLM, SGlang, dan. TensorRT-LLM

Untuk melayani beban kerja inferensi Anda menggunakan vLLM

1. Instal vLLM.

```
$ pip install vllm
```

2. Mulai server vLLM dengan NIXL. Perintah contoh berikut membuat satu instans prefill (producer) dan satu decode (konsumen) untuk koneksi handshake NIXL, konektor KV, peran KV, dan backend transportasi. Untuk contoh dan skrip terperinci, lihat Panduan Penggunaan NiXLConnector.

   Untuk menggunakan NIXL dengan EFA, atur variabel lingkungan berdasarkan pengaturan dan kasus penggunaan Anda.

   - Konfigurasi Produser (Prefiller)

```
$ vllm serve your-application \
    --port 8200 \
    --enforce-eager \
    --kv-transfer-config
 '{"kv_connector":"NixlConnector","kv_role":"kv_both","kv_buffer_device":"cuda","kv_conne
{"backends":["LIBFABRIC"]}}'
```

   - Konfigurasi Konsumen (Decoder)

```
$ vllm serve your-application \
    --port 8200 \
    --enforce-eager \
    --kv-transfer-config
 '{"kv_connector":"NixlConnector","kv_role":"kv_both","kv_buffer_device":"cuda","kv_conne
{"backends":["LIBFABRIC"]}}'
```

   Konfigurasi sampel sebelumnya menetapkan yang berikut:

- `kv_rolekekv_both`, yang memungkinkan fungsionalitas simetris di mana konektor dapat bertindak sebagai produsen dan konsumen. Ini memberikan fleksibilitas untuk pengaturan eksperimental dan skenario di mana perbedaan peran tidak ditentukan sebelumnya.
- `kv_buffer_devicekecuda`, yang memungkinkan penggunaan memori GPU.
- Backend NIXL ke`LIBFABRIC`, yang memungkinkan lalu lintas NIXL melewati EFA.

## Maksimalkan bandwidth jaringan pada instans Amazon EC2 dengan beberapa kartu jaringan

Banyak jenis instance yang mendukung EFA juga memiliki beberapa kartu jaringan. Untuk informasi selengkapnya, lihat Kartu jaringan. Jika Anda berencana menggunakan EFA dengan salah satu jenis instans ini, sebaiknya gunakan konfigurasi dasar berikut:

- Untuk antarmuka jaringan utama (indeks kartu jaringan`0`, indeks perangkat`0`), buat antarmuka ENA. Anda tidak dapat menggunakan antarmuka EFA-only jaringan sebagai antarmuka jaringan utama.
- Jika indeks kartu jaringan 0 mendukung EFA, buat antarmuka EFA-only jaringan untuk indeks kartu jaringan`0`, indeks 1 perangkat.
- Untuk setiap antarmuka jaringan tambahan, gunakan indeks kartu jaringan yang tidak digunakan berikutnya, indeks perangkat`0`, untuk antarmuka EFA-only jaringan, indeks and/or perangkat 1 untuk antarmuka jaringan ENA tergantung pada kasus penggunaan Anda, seperti persyaratan bandwidth ENA atau ruang alamat IP. Misalnya kasus penggunaan, lihatKonfigurasi EFA untuk instans P5 dan P5e.

> **Note**
>
> Instans P5 memerlukan antarmuka jaringan untuk dikonfigurasi dengan cara tertentu untuk mengaktifkan bandwidth jaringan maksimum. Untuk informasi selengkapnya, lihat Konfigurasi EFA untuk instans P5 dan P5e.

Contoh berikut menunjukkan cara meluncurkan instance berdasarkan rekomendasi ini.

Instance launch

Untuk menentukan EFA selama peluncuran instance menggunakan wizard instance peluncuran

1. Di bagian Pengaturan jaringan, pilih Edit.
2. Perluas Konfigur asi jaringan lanjutan.
3. Untuk antarmuka jaringan utama (indeks kartu jaringan`0`, indeks perangkat`0`), buat antarmuka ENA. Anda tidak dapat menggunakan antarmuka EFA-only jaringan sebagai antarmuka jaringan utama.
4. Jika indeks kartu jaringan 0 mendukung EFA, buat antarmuka EFA-only jaringan untuk indeks kartu jaringan`0`, indeks 1 perangkat.
5. Untuk setiap antarmuka jaringan tambahan, gunakan indeks kartu jaringan yang tidak digunakan berikutnya, indeks perangkat`0`, untuk antarmuka EFA-only jaringan, indeks and/or perangkat 1 untuk antarmuka jaringan ENA tergantung pada kasus penggunaan Anda, seperti persyaratan bandwidth ENA atau ruang alamat IP. Misalnya kasus penggunaan, lihatKonfigurasi EFA untuk instans P5 dan P5e.

Untuk menentukan EFA selama peluncuran instance menggunakan run-instances perintah

Untuk`--network-interfaces`, tentukan jumlah antarmuka jaringan yang diperlukan. Untuk antarmuka jaringan utama, tentukan`NetworkCardIndex=0`,`DeviceIndex=0`, dan`InterfaceType=interface`. Jika indeks kartu jaringan 0 mendukung EFA, tentukan`NetworkCardIndex=0`,`DeviceIndex=1`, dan`InterfaceType=efa-only`. Untuk antarmuka jaringan tambahan apa pun, untuk `NetworkCardIndex` menentukan indeks yang tidak digunakan berikutnya, `DeviceIndex=0` for`InterfaceType=efa-only`, and/or `DeviceIndex=1` for`InterfaceType=interface`.

Contoh cuplikan perintah berikut menunjukkan permintaan dengan 32 perangkat EFA dan satu perangkat ENA.

```
$ aws ec2 run-instances \
 --instance-type p5.48xlarge \
 --count 1 \
 --key-name key_pair_name \
 --image-id ami-0abcdef1234567890 \
 --network-interfaces
 "NetworkCardIndex=0,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
 \
```

```
"NetworkCardIndex=0,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=1,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=2,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=3,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=4,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=5,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=6,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=7,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=8,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=9,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=
only" \
"NetworkCardIndex=10,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=11,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=12,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=13,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=14,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=15,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=16,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=17,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=18,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=19,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=20,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=21,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
```

```
"NetworkCardIndex=22,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=23,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=24,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=25,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=26,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=27,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=28,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=29,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=30,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only" \
"NetworkCardIndex=31,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType
only"
...
```

Launch templates

Untuk menambahkan EFA ke template peluncuran menggunakan konsol Amazon EC2

1. Di bagian Peng aturan jaringan, perluas Konfigur asi jaringan lanjutan.
2. Untuk antarmuka jaringan utama (indeks kartu jaringan0, indeks perangkat0), buat antarmuka ENA. Anda tidak dapat menggunakan antarmuka EFA-only jaringan sebagai antarmuka jaringan utama.
3. Jika indeks kartu jaringan 0 mendukung EFA, buat antarmuka EFA-only jaringan untuk indeks kartu jaringan0, indeks 1 perangkat.
4. Untuk setiap antarmuka jaringan tambahan, gunakan indeks kartu jaringan yang tidak digunakan berikutnya, indeks perangkat0, untuk antarmuka EFA-only jaringan, indeks and/or perangkat 1 untuk antarmuka jaringan ENA tergantung pada kasus penggunaan Anda, seperti persyaratan bandwidth ENA atau ruang alamat IP. Misalnya kasus penggunaan, lihatKonfigurasi EFA untuk instans P5 dan P5e.

Untuk menambahkan EFA ke template peluncuran menggunakan buat-peluncuran-templat perintah

UntukNetworkInterfaces, tentukan jumlah antarmuka jaringan yang diperlukan. Untuk antarmuka jaringan utama, tentukanNetworkCardIndex=0,DeviceIndex=0, danInterfaceType=interface. Jika indeks kartu jaringan 0 mendukung EFA, tentukanNetworkCardIndex=0,DeviceIndex=1, danInterfaceType=efa-only. Untuk antarmuka jaringan tambahan apa pun, untuk NetworkCardIndex menentukan indeks yang tidak digunakan berikutnya, DeviceIndex=0 forInterfaceType=efa-only, and/or DeviceIndex=1 forInterfaceType=interface.

Cuplikan berikut menunjukkan contoh dengan 3 antarmuka jaringan dari kemungkinan 32 antarmuka jaringan.

```
"NetworkInterfaces":[
{
  "NetworkCardIndex":0,
  "DeviceIndex":0,
  "InterfaceType": "interface",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
{
  "NetworkCardIndex": 0,
  "DeviceIndex": 1,
  "InterfaceType": "efa-only",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
{
  "NetworkCardIndex": 1,
  "DeviceIndex": 0,
  "InterfaceType": "efa-only",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
```

```
{
  "NetworkCardIndex": 2,
  "DeviceIndex": 0,
  "InterfaceType": "efa-only",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
},
{
  "NetworkCardIndex": 3,
  "DeviceIndex": 0,
  "InterfaceType": "efa-only",
  "AssociatePublicIpAddress":false,
  "Groups":[
    "security_group_id"
  ],
  "DeleteOnTermination":true
}
...
```

Konfigurasi EFA untuk instans P5 dan P5e

`p5.48xlarge`dan `p5e.48xlarge` instance mendukung 32 kartu jaringan dan memiliki kapasitas bandwidth jaringan total 3.200 Gbps, yang hingga 800 Gbps dapat digunakan untuk lalu lintas jaringan IP. Karena lalu lintas jaringan EFA dan IP berbagi sumber daya dasar yang sama, bandwidth yang digunakan oleh satu akan mengurangi bandwidth yang tersedia untuk yang lain. Ini berarti Anda dapat mendistribusikan bandwidth jaringan antara lalu lintas EFA dan lalu lintas IP dalam kombinasi apa pun, selama total bandwidth tidak melebihi 3.200 Gbps dan bandwidth IP tidak melebihi 800 Gbps. Misalnya, jika Anda menggunakan 400 Gbps untuk bandwidth IP, Anda dapat mencapai bandwidth EFA hingga 2.800 Gbps secara bersamaan.

Gunakan kasus 1: Simpan alamat IP dan hindari potensi masalah IP Linux

Konfigurasi ini menyediakan bandwidth jaringan EFA hingga 3200 Gbps dan bandwidth jaringan IP hingga 100 Gbps dengan satu alamat IP pribadi. Konfigurasi ini juga membantu menghindari potensi masalah IP Linux, seperti penugasan otomatis alamat IP publik yang tidak diizinkan dan tantangan perutean IP (masalah pemetaan nama host ke alamat IP dan ketidakcocokan alamat IP sumber), yang dapat muncul jika sebuah instance memiliki beberapa antarmuka jaringan.

- Untuk antarmuka jaringan utama (indeks kartu jaringan 0, indeks perangkat 0), gunakan antarmuka ENA.
- Untuk indeks kartu jaringan 0, indeks perangkat 1, buat antarmuka EFA-only jaringan.
- Untuk antarmuka jaringan yang tersisa (indeks kartu jaringan 1-31, indeks perangkat 0), gunakan antarmuka EFA-only jaringan.

Kasus penggunaan 2: Bandwidth jaringan EFA dan IP maksimum

Konfigurasi ini menyediakan bandwidth jaringan EFA hingga 3200 Gbps dan bandwidth jaringan IP hingga 800 Gbps dengan 8 alamat IP pribadi. Anda tidak dapat menetapkan alamat IP publik secara otomatis dengan konfigurasi ini. Namun, Anda dapat melampirkan alamat IP Elastic ke antarmuka jaringan utama (indeks kartu jaringan 0, indeks perangkat 0) setelah peluncuran untuk konektivitas internet.

- Untuk antarmuka jaringan utama (indeks kartu jaringan 0, indeks perangkat 0), gunakan antarmuka jaringan ENA.
- Untuk antarmuka yang tersisa, lakukan hal berikut:
  - T EFA-only entukan antarmuka jaringan pada indeks kartu jaringan 0 indeks perangkat 1, dan untuk indeks kartu jaringan 1, 2, dan 3, gunakan indeks perangkat 0.
  - Tentukan satu antarmuka jaringan ENA dan empat antarmuka EFA-only jaringan di masing-masing subset indeks kartu jaringan berikut, dan gunakan indeks perangkat 1 untuk antarmuka jaringan ENA dan indeks perangkat 0 untuk antarmuka EFA-only jaringan:
    - [4,5,6,7]
    - [8,9,10,11]
    - [12,13,14,15]
    - [16.17,18,19]
    - [20,21,22,23]
    - [24.226.27]
    - [28.29.30.31]

Contoh berikut menggambarkan konfigurasi ini:

```
$ aws --region $REGION ec2 run-instances \
 --instance-type p5.48xlarge \
 --count 1 \
```

```
 --key-name key_pair_name \
 --image-id ami_id \
 --network-interfaces
 "NetworkCardIndex=0,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=0,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=1,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=2,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=3,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=4,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=4,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=5,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=6,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=7,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=8,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=8,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=9,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=10,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=11,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=12,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=12,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=13,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=14,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=15,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
```

```
"NetworkCardIndex=16,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=16,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=17,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=18,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=19,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=20,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=20,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=21,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=22,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=23,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=24,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=24,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=25,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=26,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=27,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=28,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=28,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=29,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=30,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=31,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only"
...
```

### Konfigurasi EFA untuk instance P6-B200

P6-B200 Instans memiliki total kapasitas bandwidth jaringan 3.200 Gbps, yang hingga 1600 Gbps dapat digunakan untuk ENA. Mereka memiliki 8 GPU dan 8 kartu jaringan, di mana setiap kartu jaringan mendukung bandwidth EFA hingga 400 Gbps dan bandwidth ENA 200 Gbps. Karena lalu lintas EFA dan ENA berbagi sumber daya dasar yang sama, bandwidth yang digunakan oleh satu akan mengurangi bandwidth yang tersedia untuk yang lain.

#### Gunakan kasus 1: Simpan alamat IP

Konfigurasi ini mengkonsumsi setidaknya satu alamat IP pribadi per instance dan mendukung bandwidth EFA hingga 3200 Gbps dan bandwidth ENA 200 Gbps.

- Untuk antarmuka jaringan utama (indeks kartu jaringan 0, indeks perangkat 0), gunakan antarmuka ENA.
- Untuk indeks kartu jaringan 0, indeks perangkat 1, buat antarmuka EFA-only jaringan.
- Untuk 7 kartu jaringan yang tersisa (indeks kartu jaringan 1-7, indeks perangkat 0), gunakan antarmuka EFA-only jaringan.

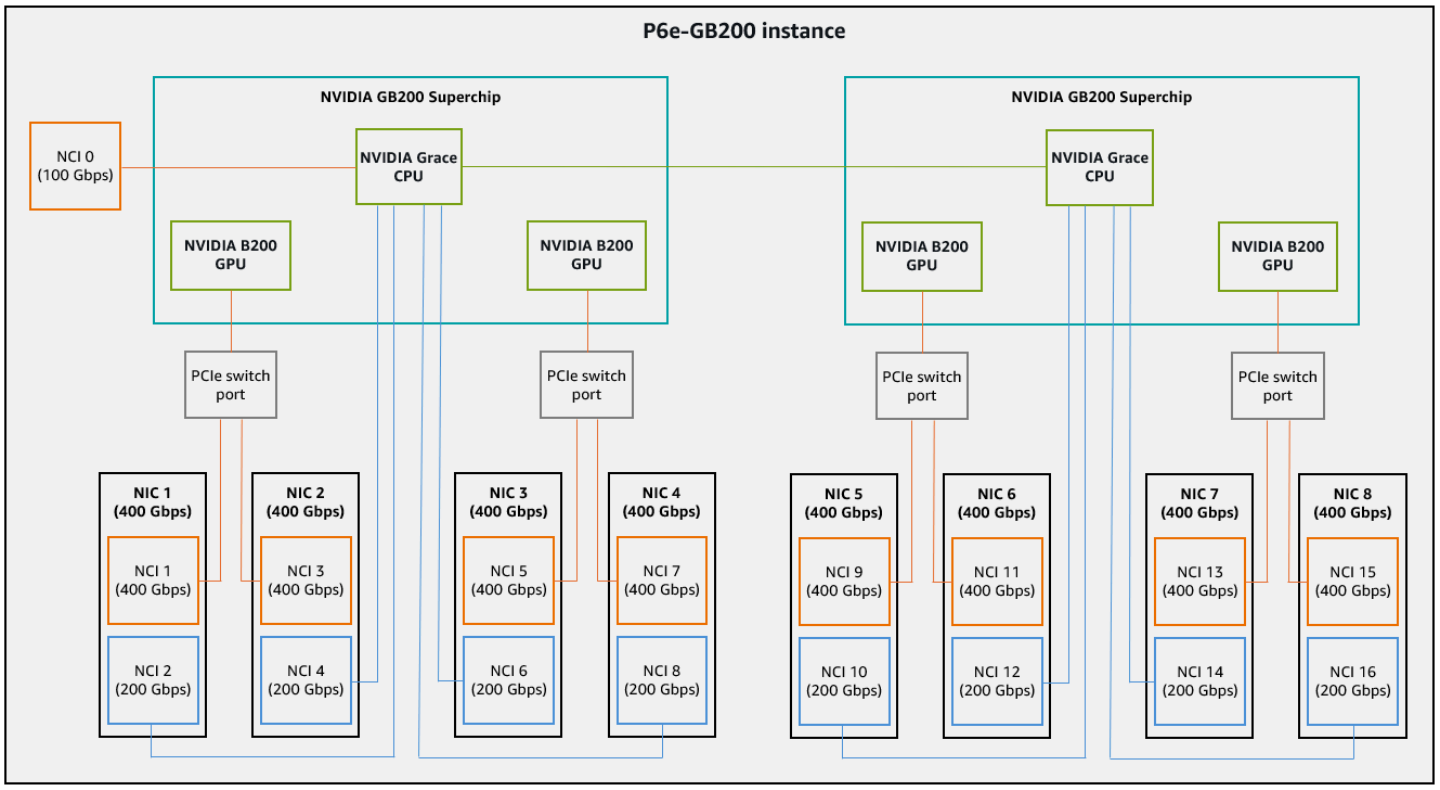
#### Kasus penggunaan 2: Bandwidth EFA dan ENA maksimum

Konfigurasi ini mengkonsumsi setidaknya 8 alamat IP pribadi per instance dan mendukung bandwidth EFA hingga 3200 Gbps dan bandwidth ENA 1600 Gbps.

- Untuk antarmuka jaringan utama (indeks kartu jaringan 0, indeks perangkat 0), gunakan antarmuka ENA.
- Untuk indeks kartu jaringan 0, indeks perangkat 1, buat antarmuka EFA-only jaringan.
- Untuk sisa 7 kartu jaringan (indeks kartu jaringan 1-7), buat antarmuka EFA-only jaringan pada indeks perangkat 0 dan antarmuka jaringan ENA pada indeks perangkat 1.

### Konfigurasi EFA untuk instance P6e-GB200

P6e-GB200 instance dapat dikonfigurasi dengan hingga 17 kartu jaringan. Gambar berikut menunjukkan tata letak kartu antarmuka jaringan fisik (NIC) untuk P6e-GB200 instance, bersama dengan pemetaan indeks kartu jaringan (NCI).




NCI utama (indeks 0) mendukung bandwidth ENA hingga 100 Gbps. NCIS dengan indeks berikut mendukung antarmuka EFA-only jaringan dan bandwidth EFA 400 Gbps: [1, 3, 5, 7, 9, 11, 13, 15]. NCIS dengan indeks berikut mendukung hingga 200 Gbps ENA atau bandwidth EFA: [2, 4, 6, 8, 10, 12, 14, 16].

NCI dalam kelompok berikut berbagi NIC fisik yang mendasarinya pada host:

- [1 dan 2]
- [3 dan 4]
- [5 dan 6]
- [7 dan 8]
- [9 dan 10]
- [11 dan 12]
- [13 dan 14]
- [15 dan 16]

Setiap NIC fisik mendukung bandwidth hingga 400 Gbps. Karena NCI dalam kelompok ini berbagi NIC fisik dasar yang sama, bandwidth yang digunakan oleh satu akan mengurangi bandwidth yang

tersedia untuk yang lain. Misalnya, jika NCI 2 menggunakan bandwidth ENA 200 Gbps, NCI 1 dapat menggunakan bandwidth EFA hingga 200 Gbps pada saat yang bersamaan.

Setiap GPU yang mendasari pada host dapat mengirim lalu lintas langsung melalui pasangan NCI berikut:

- [1 dan 3]
- [5 dan 7]
- [9 dan 11]
- [13 dan 15]

Setiap GPU mendukung bandwidth EFA hingga 400 Gbps. Karena kartu jaringan dalam grup ini berbagi GPU yang sama, bandwidth yang digunakan oleh satu akan mengurangi bandwidth yang tersedia untuk yang lain. Misalnya, jika NCI 1 menggunakan bandwidth EFA 200 Gbps, NCI 3 dapat menggunakan bandwidth EFA hingga 200 Gbps pada saat yang bersamaan. Oleh karena itu, untuk mencapai kinerja EFA maksimum, kami sarankan Anda melakukan salah satu dari hal berikut untuk mencapai total bandwidth EFA 1.600 Gbps:

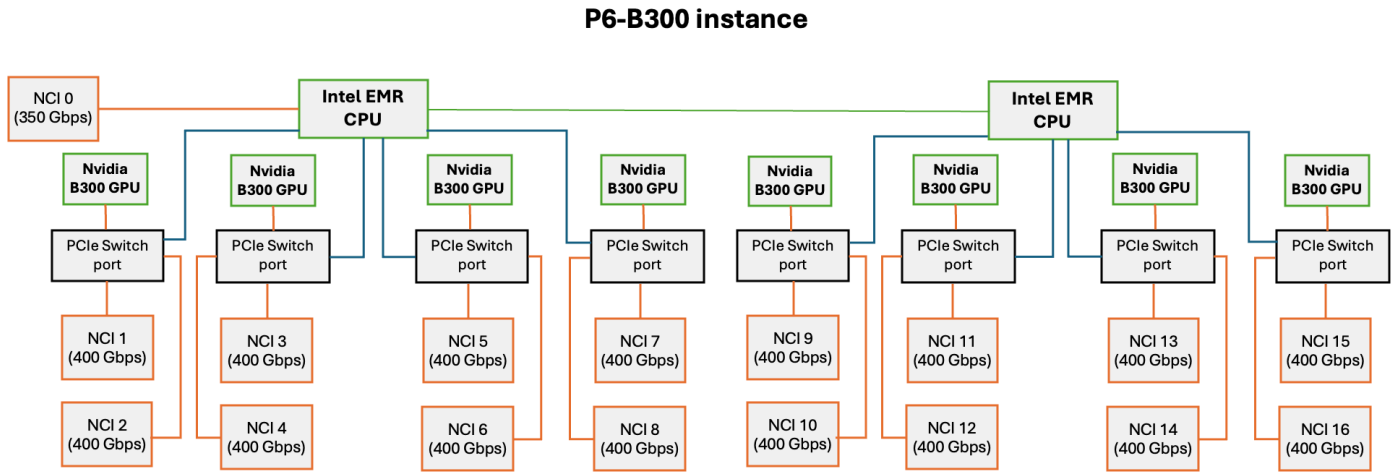
- Tambahkan antarmuka EFA-only jaringan ke hanya satu NCI di setiap grup untuk mencapai 400 Gbps per antarmuka jaringan (4 antarmuka jaringan EFA x 400 Gbps).
- Tambahkan antarmuka EFA-only jaringan ke setiap NCI di setiap grup untuk mencapai 200 Gbps per antarmuka jaringan (8 antarmuka jaringan EFA x 200 Gbps).

Misalnya, konfigurasi berikut menyediakan bandwidth EFA hingga 1.600 Gbps menggunakan antarmuka EFA-only jaringan tunggal di setiap grup NCI, dan bandwidth jaringan ENA hingga 100 Gbps hanya menggunakan NCI primer (indeks 0).

- Untuk NCI utama (indeks kartu jaringan 0, indeks perangkat 0), gunakan antarmuka jaringan ENA.
- Tambahkan antarmuka EFA-only jaringan ke berikut ini:
  - NCI 1, indeks perangkat 0
  - NCI 5, indeks perangkat 0
  - NCI 9, indeks perangkat 0
  - NCI 13, indeks perangkat 0

### Konfigurasi EFA untuk instance P6-B300

P6-B300 Instans memiliki total kapasitas bandwidth jaringan hingga 6400 Gbps untuk lalu lintas EFA, dan hingga 3870 Gbps untuk lalu lintas ENA. Mereka memiliki 8 GPU dan 17 kartu jaringan, di mana kartu jaringan utama hanya mendukung antarmuka jaringan ENA dengan bandwidth hingga 350 Gbps. Kartu jaringan sekunder mendukung hingga 400 Gbps EFA dan bandwidth ENA hingga 220 Gbps. Karena lalu lintas EFA dan ENA berbagi sumber daya dasar yang sama, bandwidth yang digunakan oleh satu akan mengurangi bandwidth yang tersedia untuk yang lain.




#### Gunakan kasus 1: Simpan alamat IP

Konfigurasi ini mengkonsumsi setidaknya satu alamat IP pribadi per instans dan mendukung bandwidth EFA hingga 6400 Gbps dan bandwidth ENA hingga 350 Gbps.

- Untuk antarmuka jaringan utama (indeks kartu jaringan 0, indeks perangkat 0), gunakan antarmuka jaringan ENA.
- Untuk kartu jaringan yang tersisa (indeks kartu jaringan 1-16, indeks perangkat 0), gunakan antarmuka EFA-only jaringan.

```
--network-interfaces \
"NetworkCardIndex=0,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=1,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=2,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
```

```
"NetworkCardIndex=3,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=4,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=5,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=6,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=7,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=8,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=9,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=10,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=11,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=12,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=13,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=14,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=15,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=16,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only"
```

Kasus penggunaan 2: Bandwidth EFA dan ENA maksimum

Konfigurasi ini mengkonsumsi setidaknya 17 alamat IP pribadi per instance dan mendukung bandwidth EFA hingga 6400 Gbps dan bandwidth ENA hingga 3870 Gbps.

- Untuk antarmuka jaringan utama (indeks kartu jaringan 0, indeks perangkat 0) gunakan antarmuka jaringan ENA.
- Untuk kartu jaringan yang tersisa, buat EFA-only antarmuka (indeks kartu jaringan 1-16 indeks perangkat 0) dan indeks kartu jaringan antarmuka ENA 1-16 indeks perangkat 1).

```
--network-interfaces \
```

```
"NetworkCardIndex=0,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=1,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=2,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=3,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=4,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=5,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=6,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=7,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=8,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=9,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=efa
only" \
"NetworkCardIndex=10,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=11,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=12,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=13,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=14,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=15,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=16,DeviceIndex=0,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=ef
only" \
"NetworkCardIndex=1,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=2,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=3,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=4,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=5,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
```

```
"NetworkCardIndex=6,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=7,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=8,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=9,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=int
 \
"NetworkCardIndex=10,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=11,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=12,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=13,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=14,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=15,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
 \
"NetworkCardIndex=16,DeviceIndex=1,Groups=security_group_id,SubnetId=subnet_id,InterfaceType=in
```

## Instal perangkat lunak EFA dalam wadah

Bagian ini mencakup cara menggunakan penginstal EFA di lingkungan kontainer. Anda dapat menginstal perangkat lunak EFA baik dengan menambahkan perintah ke Dockerfile atau dengan menjalankannya langsung di shell wadah yang sedang berjalan.

Untuk menginstal perangkat lunak EFA dalam wadah menggunakan perintah shell

1. Perbarui paket sistem operasi. Langkah ini memastikan bahwa Anda memiliki paket terbaru.
   - Untuk sistem Debian-based operasi:

     ```
     apt-get update && apt-get upgrade -y
     ```

   - Untuk sistem RHEL-based operasi:

     ```
     yum update -y
     ```

2. Unduh penginstal EFA. Kami menyarankan Anda menggunakan penginstal EFA terbaru. Untuk informasi tentang versi terbaru, lihatCatatan rilis Adaptor Kain Elastis.

```
mkdir /tmp/efa \
&& cd /tmp/efa \
&& curl -O https://efa-installer.amazonaws.com/aws-efa-installer-version.tar.gz
```

(Opsional) Verifikasi tanda tangan paket penginstal EFA.

```
wget https://efa-installer.amazonaws.com/aws-efa-installer.key \
&& gpg --import aws-efa-installer.key \
&& cat aws-efa-installer.key | gpg --fingerprint \
&& wget https://efa-installer.amazonaws.com/aws-efa-installer-version.tar.gz.sig \
&& gpg --verify ./aws-efa-installer-version.tar.gz.sig
```

3. Instal perangkat lunak EFA. Perintah berikut menggunakan argumen ini:

   - `--skip-kmod`— Melewati instalasi modul kernel. Lingkungan kontainer tidak dapat memodifikasi modul kernel.
   - `--skip-limit-conf`— Melewati konfigurasi batas. Anda tidak dapat mengatur `ulimit` di dalam wadah. Konfigurasikan ini sebagai bagian dari pembuatan wadah sebagai gantinya.
   - `--no-verify`— Melewati deteksi perangkat EFA. Deteksi perangkat menggunakan`lspci`, yang tidak berfungsi di lingkungan kontainer.

```
tar -xf aws-efa-installer-version.tar.gz \
&& cd aws-efa-installer \
&& ./efa_installer.sh -y --skip-kmod --skip-limit-conf --no-verify
```

> **Note**
>
> Untuk kontainer NGC, dengan penginstal EFA versi 1.49.0 atau yang lebih baru Anda dapat menggunakan flag alih `--enable-ngc` -alih tiga flag di atas. Bendera ini menangani semua konfigurasi khusus kontainer secara otomatis:
>
> ```
> ./efa_installer.sh -y --enable-ngc
> ```

4. Bersihkan file instalasi sementara.

```
rm -rf /tmp/efa
```

# Membuat dan memasang Adaptor Kain Elastis ke instans Amazon EC2

Anda dapat membuat EFA dan melampirkannya ke instans Amazon EC2 seperti antarmuka jaringan elastis lainnya di Amazon EC2. Namun, tidak seperti antarmuka jaringan elastis, EFA tidak dapat dilampirkan atau dilepaskan dari instance dalam `running` status.

Pertimbangan-pertimbangan

- Anda dapat mengubah grup keamanan yang terkait dengan EFA. Untuk mengaktifkan OS-bypass fungsionalitas, EFA harus menjadi anggota grup keamanan yang memungkinkan semua lalu lintas masuk dan keluar ke dan dari grup keamanan itu sendiri. Untuk informasi selengkapnya, lihat Langkah 1: Siapkan grup EFA-enabled keamanan.

  Anda mengubah grup keamanan yang terkait dengan EFA dalam cara yang sama dengan cara Anda mengubah grup keamanan yang terkait dengan antarmuka jaringan elastis. Untuk informasi selengkapnya, lihat the section called “Memodifikasi atribut antarmuka jaringan”.
- Anda menetapkan alamat IP Elastic (IPv4) dan IPv6 ke antarmuka jaringan EFA (EFA dengan ENA) dengan cara yang sama seperti Anda menetapkan alamat IP ke antarmuka jaringan elastis. Untuk informasi selengkapnya, lihat Mengelola alamat IP.

  Anda tidak dapat menetapkan alamat IP ke antarmuka EFA-only jaringan.

Tugas

- Membuat AMI
- Memasang EFA ke instans yang dihentikan
- Memasang EFA saat meluncurkan instans
- Menambahkan EFA ke templat peluncuran

## Membuat AMI

Anda dapat membuat EFA dalam subnet di VPC. Anda tidak dapat memindahkan EFA ke subnet lain dibuat, dan Anda hanya dapat memasangkannya untuk menghentikan instans dalam Zona Ketersediaan yang sama.

Console

Untuk membuat antarmuka jaringan EFA (EFA dengan ENA atau ENA-only)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Antarmuka Jaringan, lalu pilih Buat antarmuka jaringan.
3. Untuk Deskripsi, masukkan nama deskriptif untuk EFA.
4. Untuk Subnet, pilih subnet untuk membuat EFA.
5. Jenis antarmuka, pilih salah satu opsi berikut:
   - EFA dengan ENA — Untuk membuat antarmuka jaringan yang mendukung perangkat ENA dan EFA.
   - EFA-only— Untuk membuat antarmuka jaringan dengan perangkat EFA saja.
6. (Hanya untuk EFA dengan ENA) Konfigurasikan alamat IP dan penetapan awalan untuk antarmuka jaringan. Jenis alamat IP dan awalan yang dapat Anda tetapkan tergantung pada subnet yang dipilih. Untuk IPv4-only subnet, Anda dapat menetapkan alamat IP IPv4 dan awalan saja. Untuk IPv6-only subnet, Anda dapat menetapkan alamat IP IPv6 dan awalan saja. Untuk subnet dual-stack, Anda dapat menetapkan alamat IP dan awalan IPv4 dan IPv6.

   > **Note**
   >
   > Anda tidak dapat menetapkan alamat IP ke antarmuka EFA-only jaringan.

   a. Untuk alamat IPv4 Pribadi Alamat and/or IPv6, pilih Auto-assign agar Amazon EC2 secara otomatis menetapkan alamat IP dari subnet yang dipilih, atau pilih K ustom untuk menentukan alamat IP yang akan ditetapkan secara manual.

   b. Jika Anda menetapkan alamat IPv6, Anda dapat secara opsional mengaktifkan Tet apkan IP IPv6 primer. Melakukan hal ini menetapkan alamat unicast global IPv6 primer (GUA) ke antarmuka jaringan. Dengan menetapkan alamat IPv6 primer, Anda akan dapat menghindari mengganggu lalu lintas ke instans atau ENI. Untuk informasi selengkapnya, lihat alamat https://docs.aws.amazon.com/vpc/latest/userguide/vpc-ip-addressing.html#vpc-ipv6-addresses IPv6.

   c. Untuk delegasi awalan and/or IPv4 pendelegasian awalan IPv6, pilih Auto-assign agar Amazon EC2 secara otomatis menetapkan awalan dari blok CIDR subnet, atau pilih K ustom untuk menentukan awalan secara manual dari blok CIDR subnet. Jika

Anda menentukan awalan, AWS memverifikasi bahwa itu belum ditetapkan ke sumber daya lain. Untuk informasi selengkapnya, lihat Delegasi awalan untuk antarmuka jaringan Amazon EC2

d. (Opsional) Konfigurasikan pengaturan batas waktu pelacakan koneksi Idle. Untuk informasi selengkapnya, lihat Waktu habis pelacakan koneksi idle

- Batas waktu yang ditetapkan TCP — Periode batas waktu, dalam detik, untuk koneksi TCP idle dalam keadaan mapan. Min: 60 detik. Maks: 432000 detik (5 hari). Default: 432.000 detik. Direkomendasikan: Kurang dari 432000 detik.
- Batas waktu UDP — Periode batas waktu, dalam detik, untuk aliran UDP idle yang hanya melihat lalu lintas dalam satu arah atau transaksi respons permintaan tunggal. Min: 30 detik. Maks: 60 detik. Default: 30 detik.
- Waktu tunggu aliran UDP — Periode batas waktu, dalam detik, untuk aliran UDP idle yang diklasifikasikan sebagai aliran yang telah melihat lebih dari satu transaksi respons permintaan. Min: 60 detik. Maks: 180 detik (3 menit). Default: 180 detik.

7. Untuk Grup keamanan, pilih satu atau beberapa grup keamanan.
8. Pilih Buat antarmuka jaringan.

AWS CLI

Untuk membuat EFA

Gunakan perintah create-network-interface. Untuk`--interface-type`, tent`efa`ukan untuk antarmuka jaringan EFA atau`efa-only`untuk antarmuka EFA-only jaringan.

```
aws ec2 create-network-interface \
    --subnet-id subnet-0abcdef1234567890 \
    --interface-type efa \
    --description "my efa"
```

PowerShell

Untuk membuat EFA

Gunakan New-EC2NetworkInterface cmdlet. Untuk`-InterfaceType`, tent`efa`ukan untuk antarmuka jaringan EFA atau`efa-only`untuk antarmuka EFA-only jaringan

```
New-EC2NetworkInterface `
```

```
    -SubnetId subnet-0abcdef1234567890 `
    -InterfaceType efa `
    -Description "my efa"
```

## Memasang EFA ke instans yang dihentikan

Anda dapat memasang EFA ke instans yang didukung yang sedang dalam keadaan `stopped`. Anda tidak dapat memasang EFA ke instans yang sedang dalam keadaan `running`. Untuk informasi lebih lanjut tentang tipe instans yang didukung, lihat Tipe instans yang didukung.

Anda memasang EFA ke instans dengan cara yang sama seperti Anda memasang antarmuka jaringan ke suatu instans. Untuk informasi selengkapnya, lihat Lampirkan antarmuka jaringan.

## Memasang EFA saat meluncurkan instans

AWS CLI

Untuk melampirkan EFA yang ada saat meluncurkan instance

Gunakan perintah run-instance dengan opsi. `--network-interfaces` Untuk antarmuka jaringan utama, tentukan antarmuka jaringan EFA dan`NetworkCardIndex=0`,`DeviceIndex=0`. Untuk melampirkan beberapa antarmuka jaringan EFA, lihatMaksimalkan bandwidth jaringan.

```
--network-interfaces "NetworkCardIndex=0, \
    DeviceIndex=0, \
    NetworkInterfaceId=eni-1234567890abcdef0, \
    Groups=sg-1234567890abcdef0, \
    SubnetId=subnet-0abcdef1234567890"
```

Untuk melampirkan EFA baru saat meluncurkan instance

Gunakan perintah run-instance dengan opsi. `--network-interfaces` Untuk antarmuka jaringan utama, gunakan`NetworkCardIndex=0`,`DeviceIndex=0`, dan`InterfaceType=efa`. Jika Anda melampirkan beberapa antarmuka jaringan EFA, lihatMaksimalkan bandwidth jaringan.

```
--network-interfaces "NetworkCardIndex=0, \
    DeviceIndex=0, \
    InterfaceType=efa, \
    Groups=sg-1234567890abcdef0, \
    SubnetId=subnet-0abcdef1234567890"
```

PowerShell

Untuk melampirkan EFA yang ada saat meluncurkan instance

Gunakan New-EC2Instance cmdlet dengan parameter. `-NetworkInterfaces`

```
-NetworkInterface $networkInterface
```

Tentukan antarmuka jaringan sebagai berikut.

```
$networkInterface = New-Object
 Amazon.EC2.Model.InstanceNetworkInterfaceSpecification
$networkInterface.DeviceIndex = 0
$networkInterface.NetworkInterfaceId = "eni-1234567890abcdef0"
$networkInterface.Groups = @("sg-1234567890abcdef0")
$networkInterface.SubnetId = "subnet-0abcdef1234567890"
```

Untuk melampirkan EFA baru saat meluncurkan instance

Gunakan New-EC2Instance cmdlet dengan parameter. `-NetworkInterfaces`

```
-NetworkInterface $networkInterface
```

Tentukan antarmuka jaringan sebagai berikut.

```
$networkInterface = New-Object
 Amazon.EC2.Model.InstanceNetworkInterfaceSpecification
$networkInterface.DeviceIndex = 0
$networkInterface.InterfaceType = "efa"
$networkInterface.Groups = @("sg-1234567890abcdef0")
$networkInterface.SubnetId = "subnet-0abcdef1234567890"
```

## Menambahkan EFA ke templat peluncuran

Anda dapat membuat template peluncuran yang berisi informasi konfigurasi yang diperlukan untuk meluncurkan EFA-enabled instance. Anda dapat menentukan antarmuka EFA dan EFA-only jaringan di template peluncuran. Untuk membuat template EFA-enabled peluncuran, buat template peluncuran baru dan tentukan jenis instans yang didukung, EFA-enabled AMI Anda, dan grup EFA-enabled keamanan. Untuk`NetworkInterfaces`, tentukan antarmuka jaringan EFA yang akan

dilampirkan. Untuk antarmuka jaringan utama, gunakan`NetworkCardIndex=0`,`DeviceIndex=0`, dan`InterfaceType=efa`. Jika Anda melampirkan beberapa antarmuka jaringan EFA, lihatMaksimalkan bandwidth jaringan pada instans Amazon EC2 dengan beberapa kartu jaringan.

Anda dapat memanfaatkan template peluncuran untuk meluncurkan EFA-enabled instans dengan AWS layanan lain, seperti AWS Batch atau AWS ParallelCluster.

Untuk informasi lebih lanjut tentang membuat templat peluncuran, lihat Buat template peluncuran Amazon EC2.

## Melepaskan dan menghapus EFA dari instans Amazon EC2

Anda dapat melepaskan EFA dari instans Amazon EC2 dan menghapusnya dengan cara yang sama seperti antarmuka jaringan elastis lainnya di Amazon EC2.

### Melepas EFA

Untuk melepas EFA dari suatu instans, Anda harus menghentikan terlebih dahulu proses tersebut. Anda tidak dapat melepas EFA dari instans yang sedang berjalan.

Anda melepas EFA dari instans dengan cara yang sama seperti cara Anda melepas antarmuka jaringan dari suatu instans. Untuk informasi selengkapnya, lihat Lepaskan antarmuka jaringan.

### Menghapus EFA

Untuk menghapus EFA, Anda harus melepasnya terlebih dahulu dari instans. Anda tidak dapat menghapus EFA saat masih terpasang di suatu instans.

Anda menghapus EFA dengan cara yang sama seperti melihat antarmuka jaringan elastis. Untuk informasi selengkapnya, lihat Menghapus antarmuka jaringan.

## Pantau Adaptor Kain Elastis di Amazon EC2

Anda dapat menggunakan fitur-fitur berikut untuk memantau performa Elastic Fabric Adapters Anda.

Topik

- Metrik driver EFA untuk instans Amazon EC2
- Log alur Amazon VPC
- Amazon CloudWatch

# Metrik driver EFA untuk instans Amazon EC2

Driver Elastic Fabric Adapter (EFA) menerbitkan beberapa metrik dari instance yang memiliki antarmuka EFA terpasang, secara real time. Anda dapat menggunakan metrik ini untuk memecahkan masalah kinerja aplikasi dan jaringan, memilih ukuran cluster yang tepat untuk beban kerja, merencanakan aktivitas penskalaan secara proaktif, dan membandingkan aplikasi untuk menentukan apakah mereka memaksimalkan kinerja EFA yang tersedia pada instans.

Topik


- Metrik driver EFA yang tersedia
- Mengambil metrik driver EFA untuk instans Anda


## Metrik driver EFA yang tersedia

Driver EFA menerbitkan metrik berikut ke instance secara real time. Mereka memberikan jumlah kumulatif kesalahan, peristiwa koneksi, dan paket atau byte yang dikirim, diterima, ditransmisikan ulang, atau dijatuhkan oleh perangkat EFA terlampir sejak peluncuran instance atau reset driver terakhir.

| Metrik | Deskripsi | Tipe instans yang didukung |
|---|---|---|
| `tx_bytes` | Jumlah byte yang ditransmisikan.<br>Satuan: byte | Semua jenis instans yang mendukung EFA |
| `rx_bytes` | Jumlah byte yang diterima.<br>Satuan: byte | Semua jenis instans yang mendukung EFA |
| `tx_pkts` | Jumlah paket yang dikirimkan.<br>Satuan: hitung | Semua jenis instans yang mendukung EFA |
| `rx_pkts` | Jumlah paket yang diterima.<br>Satuan: hitung | Semua jenis instans yang mendukung EFA |

| Metrik | Deskripsi | Tipe instans yang didukung |
| --- | --- | --- |
| `rx_drops` | Jumlah paket yang diterima dan kemudian dijatuhkan.<br><br>Satuan: hitung | Semua jenis instans yang mendukung EFA |
| `send_bytes` | Jumlah byte yang dikirim menggunakan operasi kirim.<br><br>Satuan: byte | Semua jenis instans yang mendukung EFA |
| `recv_bytes` | Jumlah byte yang diterima oleh operasi kirim.<br><br>Satuan: byte | Semua jenis instans yang mendukung EFA |
| `send_wrs` | Jumlah paket yang dikirim menggunakan operasi kirim.<br><br>Satuan: hitung | Semua jenis instans yang mendukung EFA |
| `recv_wrs` | Jumlah paket yang diterima oleh operasi kirim.<br><br>Satuan: hitung | Semua jenis instans yang mendukung EFA |
| `rdma_write_wrs` | Jumlah operasi penulisan rdma yang diselesaikan.<br><br>Satuan: hitung | Semua jenis instans yang mendukung EFA |
| `rdma_read_wrs` | Jumlah operasi baca rdma yang diselesaikan.<br><br>Satuan: hitung | Semua jenis instans yang mendukung EFA |

| Metrik | Deskripsi | Tipe instans yang didukung |
| --- | --- | --- |
| rdma_write_bytes | Jumlah byte yang ditulis untuk itu oleh instance lain menggunakan operasi penulisan rdma.<br><br>Satuan: byte | Semua jenis instans yang mendukung EFA |
| rdma_read_bytes | Jumlah byte yang diterima menggunakan operasi baca rdma.<br><br>Satuan: byte | Semua jenis instans yang mendukung EFA |
| rdma_write_wr_err | Jumlah operasi penulisan rdma yang memiliki kesalahan lokal atau jarak jauh.<br><br>Satuan: hitung | Semua jenis instans yang mendukung EFA |
| rdma_read_wr_err | Jumlah operasi baca rdma yang memiliki kesalahan lokal atau jarak jauh.<br><br>Satuan: hitung | Semua jenis instans yang mendukung EFA |
| rdma_read_resp_bytes | Jumlah byte yang dikirim sebagai respons terhadap operasi baca rdma.<br><br>Satuan: byte | Semua jenis instans yang mendukung EFA |
| rdma_write_recv_bytes | Jumlah byte yang diterima oleh operasi penulisan rdma.<br><br>Satuan: byte | Semua jenis instans yang mendukung EFA |

| Metrik | Deskripsi | Tipe instans yang didukung |
| --- | --- | --- |
| `retrans_bytes` | Jumlah byte EFA SRD yang ditransmisikan kembali.<br><br>Satuan: hitung | Jenis instance Nitro v4 dan versi lebih baru yang mendukung EFA |
| `retrans_pkts` | Jumlah paket SRD EFA yang ditransmisikan kembali.<br><br>Satuan: byte | Jenis instance Nitro v4 dan versi lebih baru yang mendukung EFA |
| `retrans_timeout_events` | Berapa kali lalu lintas EFA SRD habis waktu dan mengakibatkan perubahan jalur jaringan.<br><br>Satuan: hitung | Jenis instance Nitro v4 dan versi lebih baru yang mendukung EFA |
| `impaired_remote_conn_events` | Jumlah kali koneksi SRD EFA memasuki keadaan terganggu , menghasilkan batas kecepatan throughput yang berkurang.<br><br>Satuan: hitung | Jenis instance Nitro v4 dan versi lebih baru yang mendukung EFA |
| `unresponsive_remote_events` | Berapa kali koneksi jarak jauh EFA SRD tidak responsif.<br><br>Satuan: hitung | Jenis instance Nitro v4 dan versi lebih baru yang mendukung EFA |

Untuk informasi selengkapnya tentang jenis instans yang mendukung EFA, lihatTipe instans yang didukung.

### Mengambil metrik driver EFA untuk instans Anda

Anda dapat menggunakan alat baris perintah rdma-tool untuk mengambil metrik untuk semua antarmuka EFA yang dilampirkan ke instance sebagai berikut:

```
$ rdma -p statistic show
link rdmap0s31/1
    tx_bytes 0
    tx_pkts 0
    rx_bytes 0
    rx_pkts 0
    rx_drops 0
    send_bytes 0
    send_wrs 0
    recv_bytes 0
    recv_wrs 0
    rdma_read_wrs 0
    rdma_read_bytes 0
    rdma_read_wr_err 0
    rdma_read_resp_bytes 0
    rdma_write_wrs 0
    rdma_write_bytes 0
    rdma_write_wr_err 0
    retrans_bytes 0
    retrans_pkts 0
    retrans_timeout_events 0
    unresponsive_remote_events 0
    impaired_remote_conn_events 0
```

Atau, Anda dapat mengambil metrik untuk setiap antarmuka EFA yang dilampirkan ke instance dari file sys menggunakan perintah berikut.

```
$ more /sys/class/infiniband/device_number/ports/port_number/hw_counters/* | cat
```

Sebagai contoh

```
$ more /sys/class/infiniband/rdmap0s31/ports/1/hw_counters/* | cat
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/lifespan
::::::::::::::
12
::::::::::::::
```

```
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_read_bytes
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_read_resp_bytes
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_read_wr_err
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_read_wrs
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_write_bytes
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_write_recv_bytes
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_write_wr_err
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rdma_write_wrs
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/recv_bytes
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/recv_wrs
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rx_bytes
::::::::::::::
0
::::::::::::::
```

```
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rx_drops
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/rx_pkts
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/send_bytes
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/send_wrs
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/tx_bytes
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/tx_pkts
::::::::::::::
0
::::::::::::::
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/retrans_bytes
::::::::::::::
0
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/retrans_pkts
::::::::::::::
0
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/retrans_timeout_events
::::::::::::::
0
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/unresponsive_remote_events
::::::::::::::
0
/sys/class/infiniband/rdmap0s31/ports/1/hw_counters/impaired_remote_conn_events
::::::::::::::
0
```

### Log alur Amazon VPC

Anda dapat membuat Amazon VPC Flow Log untuk menangkap informasi tentang lalu lintas ke dan dari EFA. Data log aliran dapat dipublikasikan ke Amazon CloudWatch Logs dan Amazon S3. Setelah membuat log alur, Anda dapat mengambil dan melihat datanya di tujuan yang dipilih. Untuk informasi selengkapnya, lihat Log Alur VPC di Panduan Pengguna Amazon VPC.

Anda membuat log alur untuk EFA dengan cara yang sama dengan cara Anda membuat log alur untuk antarmuka jaringan yang elastis. Untuk informasi selengkapnya, lihat Membuat log alur di Panduan Pengguna Amazon VPC.

Dalam entri log alur, lalu lintas EFA diidentifikasi oleh `srcAddress` dan `destAddress`, yang keduanya diformat sebagai alamat MAC, sebagaimana ditunjukkan dalam contoh berikut.

```
version accountId  eniId        srcAddress        destAddress       sourcePort destPort
 protocol packets bytes start      end        action log-status
2       3794735123 eni-10000001 01:23:45:67:89:ab 05:23:45:67:89:ab -          -
 -        9       5689  1521232534 1524512343 ACCEPT OK
```

### Amazon CloudWatch

Jika Anda menggunakan EFA di cluster Amazon EKS, Anda dapat memantau EFA Anda menggunakan Con CloudWatch tainer Insights. Amazon CloudWatch Container Insights mendukung semua metrik driver EFA, kecuali:`retrans_bytes`,,`retrans_pkts`, `retrans_timeout_eventsunresponsive_remote_events`, `danimpaired_remote_conn_events`.

Untuk informasi selengkapnya, lihat metrik Amazon EKS dan Kubernetes Container Insights di Panduan Pengguna Amazon CloudWatch .

## Memverifikasi penginstal EFA menggunakan checksum

Anda dapat secara opsional memverifikasi tarball EFA (`.tar.gz`file) menggunakan checksum MD5 atau SHA256. Kami menyarankan Anda melakukan hal ini untuk memverifikasi identitas penerbit perangkat lunak dan memeriksa apakah aplikasi tersebut belum diubah atau rusak sejak file tersebut diterbitkan.

Untuk memverifikasi tarball

Gunakan utilitas md5sum untuk checksum MD5, atau utilitas sha256sum untuk checksum SHA256, dan tentukan nama file tarball. Anda harus menjalankan perintah dari direktori tempat Anda menyimpan file tarball.

- MD5*

```
$  md5sum tarball_filename.tar.gz
```

- SHA256

```
$  sha256sum tarball_filename.tar.gz
```

Perintah tersebut harus mengembalikan nilai checksum dalam format berikut.

```
checksum_value tarball_filename.tar.gz
```

Bandingkan nilai checksum yang dikembalikan oleh perintah dengan nilai checksum yang diberikan dalam tabel di bawah ini. Jika checksum cocok, skrip penginstalan dapat dijalankan dengan aman. Jika checksum tidak cocok, jangan jalankan skrip instalasi, dan hubungi Dukungan.

Misalnya, perintah berikut memverifikasi tarball EFA 1.9.4 menggunakan checksum SHA256.

```
$  sha256sum aws-efa-installer-1.9.4.tar.gz
```

Contoh output:

```
1009b5182693490d908ef0ed2c1dd4f813cc310a5d2062ce9619c4c12b5a7f14 aws-efa-
installer-1.9.4.tar.gz
```

Tabel berikut mencantumkan checksum untuk versi EFA terbaru.

| Versi | Checksum |
| --- | --- |
| EFA 1.50.0 | MD5: e5a5178944b1f1112f3b2eb3b15ca5a7 |

| Versi | Checksum |
| --- | --- |
| | SHA256: fa6dff8593d866866c 13cb4640d9059835cd4efa42797 1f100ab40c97bef2841 |
| EFA 1.49.0 | MD5: bb10de6856e08a7e7e8a6b52bea 6f681<br><br>SHA256: cf2e9281a2328a243c 76f911a490faed43ca0fecfe473 3c25e34b2e92a32c309 |
| EFA 1.48,0 | MD5: d30960f1876fb16ac8238193a26 c6b29<br><br>SHA256: f3a8aadc2b94288694 a9b9375bcad84a82f0f843e0eef afda7b158b2200549cc |
| EFA 1.47,0 | MD5: c81d4caf24dabc04a6e48185906 20f5f<br><br>SHA256: 2df4201e046833c7dc 8160907bee7f52b76ff80ed1473 76a2d0ed8a0dd66b2db |
| EFA 1.46,0 | MD5: a88bbd9b71624d7ca401b54bc2fc0c19<br><br>SHA256: 8302bd7849afb95c90 3a875d7dcb6f85b3d7629e9a8b6 7d020031cfc6f4d0ee1 |
| EFA 1.45,1 | MD5: 91c3c87e16bbcaca1513252c38b771bb<br><br>SHA256: 9aeb20c645135b6039 cc08986d8f14e63280f7839e882 a74df5e83627ffeaa17 |

| Versi | Checksum |
| --- | --- |
| EFA 1.45,0 | MD5:800aeddfa9d9c5f139a7b8f7c4fec627<br><br>SHA256: 25ba26a0877fe33173 90dc126aad2f23e27fc461cf0b9 40004f032cb342fa539 |
| EFA 1.44.0 | MD5: d024f6bebe080db42745103b84ca7c43<br><br>SHA256: f129a5b44a49d593d2 47e55a59eb9bcb57121566e1c2e 42b832a4e794fa83d8a |
| EFA 1,43,3 | MD5: 4dbc6eeecc516760253c10cbedb 6319d<br><br>SHA256: 6c470ebce254c71653 47b5048895ac2996c8856727164 2297f4c597738300652 |
| EFA 1,43,2 | MD5: 7287b25a07c9747c0d4001e8fc5 f59b2<br><br>SHA256: de15c5bdbc83b952af bde876110830c604ad0796680e5 157c05f7c1979a41069 |
| EFA 1.43.1 | MD5: 7cfafc8debaea51dd4966fa0b2b ba673<br><br>SHA256: 54211eda0c193138ee8ed09b5fb 41c41fc76fe0a77935fa4ec8d989466342740 |

| Versi | Checksum |
| --- | --- |
| EFA 1.43.0 | MD5: f2b3dd7dc8670b541f7c23fd58e5e503<br><br>SHA256: 786df3458c499237be33bb8e50ffd4da7c18c20e254380ffc80fb90833d8cc73 |
| EFA 1,42,0 | MD5: 94b2b1db09da1dde08ec049db1f24370<br><br>SHA256: 4114fe612905ee05083ae5cb391a00a012510f3abfecc642d86c9a5ae4be9008 |
| EFA 1.41.0 | MD5: 086181c3ee3f8da512fc6e1c795e8936<br><br>SHA256: 3506354cdfbe31ff552fe75f5d0d9bb7efd29cf79bd99457347d29c751c38f9f |
| EFA 1,40,0 | MD5: f3ec6f73fbeaccba082327507581157c<br><br>SHA256: 30491b0fe7c3470d4439594538855c981b05fa69862d74f8c05eb9b97912368a |
| EFA 1.39.0 | MD5: c223d5954a85a7fbcd248c942b866e43<br><br>SHA256: 2cbc028c03064633bb990782b47c36156637769e2f48704417a9c700a7a32101 |

| Versi | Checksum |
| --- | --- |
| EFA 1,38,1 | MD5: f112569e828ab65187777f794bab542c<br><br>SHA256: 83923374afd388b1cfcf4b3a21a2b1ba7cf46a01a587f7b519b8386cb95e4f81 |
| EFA 1,38,0 | MD5: 43a2a446b33a2506f40853d55059f1ea<br><br>SHA256: 4f436954f35ad53754b4d005fd8d0be63de3b4184de41a695b504bdce0fecb22 |
| EFA 1.37.0 | MD5: 6328070192bae920eca45797ad4c1db1<br><br>SHA256: 2584fc3c8bb99f29b3285e275747ff09d67c18e162c2a652e36c976b72154bfb |
| EFA 1.36.0 | MD5: 1bec83180fbffb23452ab6469ca21dfa<br><br>SHA256: de183f333cfb58aeb7908a67bf9106985ba3ccb7f8638b851d2a0d8dbfacaec4 |
| EFA 1.35.0 | MD5: 252f03c978dca5f8e8d9f34e488b256e<br><br>SHA256: 432b6ad4368ba0cd8b902729d14a908a97be7a3dcc5239422ea994a47f35a5e1 |

| Versi | Checksum |
| --- | --- |
| EFA 1.34.0 | MD5: 5cd4b28d27a31677c16139b54c9acb45<br><br>SHA256: bd68839e741b0afd3ec2e37d50603803cfa7a279c120f0a736cc57c2ff2d7fdc |
| EFA 1.33.0 | MD5: e2f61fccbcaa11e2ccfddd3660522276<br><br>SHA256: 0372877b87c6a7337bb7791d255e1053b907d030489fb2c3732ba70069185fce |
| EFA 1,32,0 | MD5: db8d65cc028d8d08b5a9f2d88881c1b1<br><br>SHA256: 5f7233760be57f6fee6de8c09acbfbf59238de848e06048dc54d156ef578fc66 |
| EFA 1.31.0 | MD5: 856352f12bef2ccbadcd75e35aa52aaf<br><br>SHA256: 943325bd37902a4300ac9e5715163537d56ecb4e7b87b37827c3e547aa1897bf |
| EFA 1.30.0 | MD5: 31f48e1a47fe93ede8ebd273fb747358<br><br>SHA256: 876ab9403e07a0c3c91a1a34685a52eced890ae052df94857f6081c5f6c78a0a |

| Versi | Checksum |
| --- | --- |
| EFA 1.29.1 | MD5: e1872ca815d752c1d7c2b5c175e52a16<br><br>SHA256: 178b263b8c25845b63dc93b25bcdff5870df5204ec509af26f43e8d283488744 |
| EFA 1.29.0 | MD5: 39d06a002154d94cd982ed348133f385<br><br>SHA256: 836655f87015547e733e7d9f7c760e4e24697f8bbc261bb5f3560abd4206bc36 |
| EFA 1.28.0 | MD5: 9dc13b7446665822605e66febe074035<br><br>SHA256: 2e625d2d6d3e073b5178e8e861891273d896b66d03cb1a32244fd56789f1c435 |
| EFA 1.27.0 | MD5: 98bfb515ea3e8d93f554020f3837fa15<br><br>SHA256: 1d49a97b0bf8d964d91652a79ac851f2550e33a5bf9d0cf86ec9357ff6579aa3 |
| EFA 1.26.1 | MD5: 884e74671fdef4725501f7cd2d451d0c<br><br>SHA256: c616994c924f54ebfabfab32b7fe8ac56947fae00a0ff453d975e298d174fc96 |

| Versi | Checksum |
| --- | --- |
| EFA 1.26.0 | MD5: f8839f12ff2e3b9ba09ae8a82b30e663<br><br>SHA256: bc1abc1f76e97d204d3755d2a9ca307fc423e51c63141f798c2f15be3715aa11 |
| EFA 1.25.1 | MD5: 6d876b894547847a45bb8854d4431f18<br><br>SHA256: d2abc553d22b89a4ce92882052c1fa6de450d3a801fe005da718b7d4b9602b06 |
| EFA 1.25.0 | MD5: 1993836ca749596051da04694ea0d00c<br><br>SHA256: 98b7b26ce031a2d6a93de2297cc71b03af647194866369ca53b60d82d45ad342 |
| EFA 1.24.1 | MD5: 211b249f39d53086f3cb0c07665f4e6f<br><br>SHA256: 120cfeec233af0955623ac7133b674143329f9561a9a8193e473060f596aec62 |
| EFA 1.24.0 | MD5: 7afe0187951e2dd2c9cc4b572e62f924<br><br>SHA256: 878623f819a0d9099d76ecd41cf4f569d4c3aac0c9bb7ba9536347c50b6bf88e |

| Versi | Checksum |
| --- | --- |
| EFA 1.23.1 | MD5: 22491e114b6ee7160a8290145dca0c28<br><br>SHA256: 5ca848d8e0ff4d1571cd443c36f8d27c8cdf2a0c97e9068ebf000c303fc40797 |
| EFA 1.23.0 | MD5: 38a6d7c1861f5038dba4e441ca7683ca<br><br>SHA256: 555d497a60f22e3857fdeb3dfc53aa86d05926023c68c916d15d2dc3df6525bd |
| EFA 1.22.1 | MD5: 600c0ad7cdbc06e8e846cb763f92901b<br><br>SHA256: f90f3d5f59c031b9a964466b5401e86fd0429272408f6c207c3f9048254e9665 |
| EFA 1.22.0 | MD5: 8f100c93dc8ab519c2aeb5dab89e98f8<br><br>SHA256: f329e7d54a86a03ea51da6ea9a5b68fb354fbae4a57a02f9592e21fce431dc3a |
| EFA 1.21.0 | MD5: 959ccc3a4347461909ec02ed3ba7c372<br><br>SHA256: c64e6ca34ccfc3ebe8e82d08899ae8442b3ef552541cf5429c43d11a04333050 |

| Versi | Checksum |
| --- | --- |
| EFA 1.20.0 | MD5: `7ebfbb8e85f1b94709df4ab3db47913b`<br><br>SHA256: `aeefd2681ffd5c4c631d1502867db5b831621d6eb85b61fe3ec80df983d1dcf0` |
| EFA 1.19.0 | MD5: `2fd45324953347ec5518da7e3fefa0ec`<br><br>SHA256: `99b77821b9e72c8dea015cc92c96193e8db307deee05b91a58094cc331f16709` |
| EFA 1.18.0 | MD5: `fc2571a72f5d3c7b7b576ce2de38d91e`<br><br>SHA256: `acb18a0808aedb9a5e485f1469225b9ac97f21db9af78e4cd6939700debe1cb6` |
| EFA 1.17.3 | MD5: `0517df4a190356ab559235147174cafd`<br><br>SHA256: `5130998b0d2883bbae189b21ab215ecbc1b01ae0231659a9b4a17b0a33ebc6ca` |
| EFA 1.17.2 | MD5: `a329dedab53c4832df218a24449f4c9a`<br><br>SHA256: `bca1fdde8b32b00346e175e597ffab32a09a08ee9ab136875fb38283cc4cd099` |

| Versi | Checksum |
| --- | --- |
| EFA 1.17.1 | MD5: `733ae2cfc9d14b52017eaf0a2ab6b0ff`<br><br>SHA256: `f29322640a88ae9279805993cb836276ea240623820848463ca686c8ce02136f` |
| EFA 1.17.0 | MD5: `d430fc841563c11c3805c5f82a4746b1`<br><br>SHA256: `75ab0cee4fb6bd38889dce313183f5d3a83bd233e0a6ef6205d8352821ea901d` |
| EFA 1.16.0 | MD5: `399548d3b0d2e812d74dd67937b696b4`<br><br>SHA256: `cecec36495a1bc6fdc82f97761a541e4fb6c9a3cbf3cfcb145acf25ea5dbd45b` |
| EFA 1.15.2 | MD5: `955fea580d5170b05823d51acde7ca21`<br><br>SHA256: `84df4fbc1b3741b6c073176287789a601a589313accc8e6653434e8d4c20bd49` |
| EFA 1.15.1 | MD5: `c4610267039f72bbe4e35d7bf53519bc`<br><br>SHA256: `be871781a1b9a15fca342a9d169219260069942a8bda7a8ad06d4baeb5e2efd7` |

| Versi | Checksum |
| --- | --- |
| EFA 1.15.0 | MD5: 9861694e1cc00d884fadac07d22898be<br><br>SHA256: b329862dd5729d2d098d0507fb486bf859d7c70ce18b61c302982234a3a5c88f |
| EFA 1.14.1 | MD5: 50ba56397d359e57872fde1f74d4168a<br><br>SHA256: c7b1b48e86fe4b3eaa4299d3600930919c4fe6d88cc6e2c7e4a408a3f16452c7 |
| EFA 1.14.0 | MD5: 40805e7fd842c36ececb9fd7f921b1ae<br><br>SHA256: 662d62c12de85116df33780d40e0533ef7dad92709f4f613907475a7a1b60a97 |
| EFA 1.13.0 | MD5: c91d16556f4fd53becadbb345828221e<br><br>SHA256: ad6705eb23a3fce44af3afc0f7643091595653a723ad0374084f4f2b715192e1 |
| EFA 1.12.3 | MD5: 818aee81f097918cfaebd724eddea678<br><br>SHA256: 2c225321824788b8ca3fbc118207b944cdb096b847e1e0d1d853ef2f0d727172 |

| Versi | Checksum |
| --- | --- |
| EFA 1.12.2 | MD5: `956bb1fc5ae0d6f0f87d2e481d49fccf`<br><br>SHA256: `083a868a2c212a5a4fcf3e4d732b685ce39cceb3ca7e5d50d0b74e7788d06259` |
| EFA 1.12.1 | MD5: `f5bfe52779df435188b0a2874d0633ea`<br><br>SHA256: `5665795c2b4f09d5f3f767506d4d4c429695b36d4a17e5758b27f033aee58900` |
| EFA 1.12.0 | MD5: `d6c6b49fafb39b770297e1cc44fe68a6`<br><br>SHA256: `28256c57e9ecc0b0778b41c1f777a9982b4e8eae782343dfe1246079933dca59` |
| EFA 1.11.2 | MD5: `2376cf18d1353a4551e35c33d269c404`<br><br>SHA256: `a25786f98a3628f7f54f7f74ee2b39bc6734ea9374720507d37d3e8bf8ee1371` |
| EFA 1.11.1 | MD5: `026b0d9a0a48780cc7406bd51997b1c0`<br><br>SHA256: `6cb04baf5ffc58ddf319e956b5461289199c8dd805fe216f8f9ab8d102f6d02a` |

| Versi | Checksum |
| --- | --- |
| EFA 1.11.0 | MD5: 7d9058e010ad65bf2e14259214a36949<br><br>SHA256: 7891f6d45ae33e822189511c4ea1d14c9d54d000f6696f97be54e915ce2c9dfa |
| EFA 1.10.1 | MD5: 78521d3d668be22976f46c6fecc7b730<br><br>SHA256: 61564582de7320b21de319f532c3a677d26cc46785378eb3b95c636506b9bcb4 |
| EFA 1.10.0 | MD5: 46f73f5a7afe41b4bb918c81888fefa9<br><br>SHA256: 136612f96f2a085a7d98296da0afb6fa807b38142e2fc0c548fa986c41186282 |
| EFA 1.9.5 | MD5: 95edb8a209c18ba8d250409846eb6ef4<br><br>SHA256: a4343308d7ea4dc943ccc21bcebed913e8868e59bfb2ac93599c61a7c87d7d25 |
| EFA 1.9.4 | MD5: f26dd5c350422c1a985e35947fa5aa28<br><br>SHA256: 1009b5182693490d908ef0ed2c1dd4f813cc310a5d2062ce9619c4c12b5a7f14 |

| Versi | Checksum |
| --- | --- |
| EFA 1.9.3 | MD5: `95755765a097802d3e6d5018d1a5d3d6`<br><br>SHA256: `46ce732d6f3fcc9edf6a6e9f9df0ad136054328e24675567f7029edab90c68f1` |
| EFA 1.8.4 | MD5: `85d594c41e831afc6c9305263140457e`<br><br>SHA256: `0d974655a09b213d7859e658965e56dc4f23a0eee2dc44bb41b6d039cc5bab45` |

## Catatan rilis Adaptor Kain Elastis

Tabel berikut menjelaskan riwayat versi dan changelog untuk perangkat lunak Elastic Fabric Adapter.

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.50.0 | • Hentikan dukungan untuk Amazon Linux 2. Versi installer EFA terakhir yang mendukung AL2 adalah 1.49.0.<br>• Tingkatkan ke libfabric 2.6.0amzn1.0<br>• Tingkatkan ke driver EFA 3.3.0<br>• Tingkatkan ke rdma-core 64.0<br>• Tingkatkan ke Plu AWS gin OFI NCCL 1.21.1 | Agustus 21, 2026 |
| 1.49.0 | • Tingkatkan ke libfabric 2.4.0amzn5.0<br>• Tingkatkan ke driver EFA 3.1.0 | Juni 27, 2026 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| | • Tingkatkan ke rdma-core 63.0<br>• Tingkatkan ke Plugin AWS OFI NCCL 1.20.0<br>• Hentikan dukungan untuk openSUSE Leap.<br>• Pembaruan instal NGC dan perbaikan bug:<br>  • Instal bin/dev paket libfabric-aws- yang hilang saat `--build-ngc` digunakan.<br>  • Hapus deteksi otomatis kasus penggunaan build-NGC, dan hapus opsi terkait. `--disable-build-ngc`<br>  • `--skip-rdma-core` tidak lagi tersirat ketika `--build-ngc` ditentukan.<br>  • Perbaikan bug: saat menggunakan `--build-ngc` dan tidak `--skip-rdma-core` ditentukan, rdma-core sekarang diinstal seperti yang diharapkan.<br>  • Instalasi penuh dihindari di semua kasus penggunaan instalasi NGC.<br>  • `--skip-mpi` dan `--skip-rdma-core` sekarang menjadi pilihan global, tidak terbatas pada kasus penggunaan NGC. | |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.48.0 | • Tingkatkan ke libfabric 2.4.0amzn3.0<br>• Optimalkan pengambilan alamat shm dalam operasi RDM<br>• Perbaiki bug penggunaan setelah bebas dalam penyelesa ian paket DC, pembuangan CQ, uji unit AH, dan pencatatan MR<br>• Perbaiki kondisi balapan di efa_rdm_ep_get_peer dan penguncian QP<br>• Perbaiki array deskriptor yang tidak diinisialisasi yang menyebabkan segfault dengan FI_EFA_ENABLE_SHM_ TRANSFER=0<br>• Perbaiki kebocoran memori di fi_info, err_buf, penghancu ran perangkat EFA, dan pengujian unit<br>• Tambahkan pemeriksaan null untuk base_ep dalam polling CQ untuk menangani QP yang hancur<br>• Setel qp->base_ep di efa_qp_create dan kembalikan kesalahan jika efa_mr_reg_ibv_mr mengembalikan NULL<br>• Pindahkan tabel QP dari domain ke perangkat dengan penguncian tingkat perangkat<br>• Tambahkan pembuatan QP ke ID permintaan langsung jalur data dan verifikasi nomor QP di CQE<br>• Berhenti menyetel qp->ibv_qp_ex->wr_id untuk jalur data langsung<br>• Hapus txe dengan benar dari daftar longcts_send pada penyelesaian tanda terima | April 14, 2026 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| | • Refaktor dan bersihkan efa_cq_handle_error<br>• Ubah penghitung registrasi MR menjadi operasi atom<br>• Migrasikan kode pendaftaran SHM MR dan perkenalkan fungsi mr regv internal<br>• Nonaktifkan cache MR secara default di bawah ASAN<br>• Hanya menonaktifkan zcpy_rx ketika p2p tidak tersedia tetapi FI_HMEM diminta<br>• Jangan lacak tx pkt pool untuk build non-debug<br>• Tingkatkan prosedur wait_send dan fungsi tiruan<br>• Sesuaikan level log untuk efa-direct CQ err_data dan efa_show_help<br>• Tentukan makro sanitizer dan nonaktifkan memhook di bawah ASAN<br>• Tambahkan pengujian unit untuk ep aktifkan jalur kesalahan dan penanganan penyelesaian kesalahan<br>• Tingkatkan ke libnccl-ofi 1.19.0<br>• Memperbaiki kesalahan dalam menangani pegangan dma-buf yang dibuat oleh ROCm/HSA dan menambahkan deteksi dukungan dma-buf saat menggunakan runtime HSA pada platform RoCM<br>• Memperbaiki potensi kebocoran memori<br>• | |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| | Memperbaiki pembuatan topologi NCCL untuk GB200 dalam wadah Docker di mana node NUMA terputus dari node Paket menyebabkan pembuatan topologi yang salah<br>• Mengurangi pemanfaatan QP pada NIC 0 selama inisialis asi<br>• Pilihan algoritma tuner yang ditingkatkan untuk P6-B200 dan jenis P6-B300 instance<br>• Dukungan untuk instalasi libnccl-ofi pada wadah NGC<br>• Tambahkan dukungan verifikasi tanda tangan RPM/DEB paket individual<br>• Tambahkan dukungan untuk RHEL 10<br>• Tambahkan Plugin OFI NCCL pada Debian 12 dan RHEL 10 | |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.47.0 | • Tingkatkan ke libfabric 2.4.0amzn1.0<br>  • Refaktor manajemen rekan: pindahkan titik akhir → peer hashmap ke level AV<br>  • Tambahkan penghitung pembuatan entri paket untuk mencegah masalah ABA<br>  • Terapkan pemblokiran dukungan baca CQ (fi_cq_sread) dengan objek tunggu<br>  • Perbaiki double-free di eager/mulreq /longread pemrosesan paket RTM<br>  • Perbaiki bendera akses IBV pendaftaran MR berdasarkan kemampuan RDMA perangkat<br>  • Nonaktifkan FI_OPT_EFA_SENDRECV_IN_ORDER_ALIGNED_128_BYTES (mengembalikan -FI_EOPNOTSUPP)<br>  • Optimalkan entri paket agar sesuai dengan dua baris cache x86 (128 byte)<br>  • Selalu berhasil efa_rdm_srx_start dan geser jendela recv pada kesalahan RTM/RTA<br>  • Putuskan tautan entri paket RX sebelum dirilis selama pembersihan titik akhir<br>• Tingkatkan ke libnccl-ofi 1.18.0<br>  • P6-B300 dukungan: menambahkan keputusan tuner khusus untuk P6-B300<br>  • | Januari 29, 2026 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| | Peningkatan kinerja PAT P6-B200 dengan mengurangi jumlah saluran untuk ukuran pesan yang lebih kecil<br>• Default protokol yang diubah, default ke protokol RDMA di Trn1 dan default ke protokol SENDRECV pada g7e.8xlarge<br>• Pemilihan platform dinamis: fitur tambahan untuk mengaktifkan AWS pengoptimalan saat runtime berdasarkan keberadaan NIC AWS . Ini memungkinkan biner plugin tunggal untuk digunakan untuk kedua AWS dan non-AWS platform. AWS pengoptimalan masih dapat dinonaktifkan pada waktu kompilasi.<br>• Dukungan tetap untuk penyedia non-FI_mr_virt_ADDR dalam protokol RDMA<br>• Peningkatan kecepatan tautan NIC PCIe dan pelaporan lebar ke NCCL<br>• Model threading yang didesain ulang untuk mendukung aplikasi multi-utas tanpa memerlukan domain Libfabric terpisah untuk setiap thread.<br>• Dukungan tetap untuk penyedia FI_MR_ENDPOINT (hanya mendukung protokol SENDRECV) dengan membersihkan sumber daya dalam urutan yang benar<br>• Tingkatkan ke rdma-core 61.0<br>• Gunakan CQ berulir tunggal jika Domain Thread disediakan<br>• Verifikasi nomor QP pada proses CQE<br>• Tingkatkan ke driver efa 3.0.0<br>• Meningkatkan penanganan kesalahan admin<br>• | |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| | Periksa kebenaran nomor QP pada penyelesaian jajak pendapat<br>• Hapus backport tanpa syarat dari pencarian ukuran halaman terbaik<br>• Cetak string errno untuk pointer kesalahan<br>• Sederhanakan kode dalam alur pembuatan CQ<br>• Backport perubahan hulu di CQ dengan aliran pembuatan umem<br>• Sesuaikan konfigurasi DKMS untuk versi DKMS baru<br>• Tingkatkan ke Open MPI 5.0.9amzn1<br>• Bugfix: Atur level threading domain berdasarkan dukungan utas MPI<br>• Bugfix: Minta tanda FI_COMPLETATION untuk memastikan penyelesaian dihasilkan untuk semua peristiwa transfer data<br>• Bagikan domain antara BTL dan MTL untuk mengurangi jumlah total domain yang dibuat untuk memperbaiki kehabisan sumber daya pada sistem dengan jumlah inti tinggi | |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.46.0 | • Tambahkan Dukungan untuk Debian 13<br>• Tingkatkan ke efa-nv-peermem 1.2.3<br>  • Perbaiki kesalahan dan peringatan build dengan kernel dan DKMS baru<br>• Tingkatkan ke libfabric 2.3.1amzn4.0<br>  • Aktifkan jalur data langsung untuk jalur protokol efa-rdm<br>  • Aktifkan dukungan RoCR HMEM<br>  • Opsi FI_OPT_EFA_USE_UNSOLICITED_WRITE_RECV baru untuk menonaktifkan recv tulis yang tidak diminta<br>  • Optimalkan waktu pembentukan koneksi<br>• Tambahkan libnccl-ofi di RHEL 9, Rocky Linux 9, dan Debian 13 | Desember 12, 2025 |
| 1.45.1 | • Tingkatkan ke libfabric 2.3.1amzn3.0<br>  • Perbaiki bug yang memotong cq_data menjadi 2 byte sementara penyedia mengiklankan dukungan untuk 4 byte | November 26, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.45.0 | • Tingkatkan ke rdma-core 60.amzn0<br>  • Perpanjang kueri DV CQ untuk mengembalikan bel pintu<br>• Tingkatkan ke libfabric 2.3.1amzn2.0<br>  • Mendukung mode FI_RX_CQ_DATA untuk efa-direct<br>  • Optimalkan posting WQE di jalur langsung jalur data<br>  • Perbaiki kondisi balapan dalam proses cntr<br>  • Tingkatkan logging untuk pesan peringatan<br>  • Menangani penyelesaian RMA dengan benar untuk rekan yang dihapus<br>  • Usir entri AH dari AV implisit saat batas AH tercapai<br>• Tingkatkan ke libnccl-ofi 1.17.2<br>  • Memperbaiki kerusakan dengan NCCL v2.28.x saat inisialis asi Libfabric gagal<br>  • Menambahkan dukungan untuk keluarga instance g7e<br>  • Memperbaiki masalah di mana NCCL dapat secara keliru mencoba menggunakan jalur RDMA GPUDirect pada platform yang mendukung DMA-BUF<br>  • Memperbaiki masalah pemesanan shutdown pada NIC yang memerlukan pendaftaran memori per titik akhir (Cray Slingshot) | November 17, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.44.0 | • Tingkatkan ke rdma-core 59.amzn0<br>  • Tambahkan dukungan untuk mengikat QP dan CQ ke domain utas<br>  • Tambahkan dukungan untuk mengalokasikan domain induk<br>  • Tambahkan dukungan untuk mengalokasikan domain thread<br>  • Tambahkan kueri kata kerja langsung QP dan CQ<br>  • Tambahkan opsi untuk membuat CQ dengan memori eksternal`<br>  • Perbaiki polling CQ setelah penghancuran QP<br>• Tingkatkan ke libfabric 2.3.1amzn1.0<br>• Tingkatkan ke Open MPI 5.0.8amzn1<br>  • Peningkatan fungsionalitas<br>  • Perbaikan bug<br>• Tingkatkan ke libnccl-ofi 1.17.1<br>  • Mendukung NCCL v2.28.3-1 sambil mempertahankan kompatibilitas mundur dengan NCCL v2.17.1 dan yang lebih baru.<br>  • Membangun dengan platform-aws membutuhkan Libfabric v1.22.0amzn4.0 atau lebih tinggi. Diuji dengan versi hingga Libfabric v2.3.1amzn1.0.<br>  • | Oktober 29, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| | Perbaikan untuk mendukung kompatibilitas di CUDA12 dan CUDA13. | |
| 1.43.3 | • Tingkatkan ke driver efa 2.17.3<br>• Mendukung P2P dengan driver NVIDIA 580<br>• Sesuaikan registrasi MR untuk 6.17 dan pada kernel | Oktober 01, 2025 |
| 1.43.2 | • Tingkatkan ke libnccl-ofi 1.16.3<br>• Mendukung NCCL v2.27.7-1 sambil mempertahankan kompatibilitas mundur dengan NCCL v2.17.1 dan yang lebih baru.<br>• Membangun dengan platform-aws membutuhkan Libfabric v1.22.0amzn4.0 atau lebih tinggi. Diuji dengan versi hingga Libfabric 2.1.0amzn5.0.<br>• Aktifkan domain-per-thread secara default pada semua jenis instans Amazon EC2 untuk meningkatkan kinerja untuk beberapa aplikasi di mana NCCL membuat beberapa utas proxy | Agustus 15, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.43.1 | • Tingkatkan ke libnccl-ofi 1.16.2<br>  • Mendukung NCCL v2.27.6-1 sambil mempertahankan kompatibilitas mundur dengan NCCL v2.17.1 dan yang lebih baru.<br>  • Membangun dengan platform-aws membutuhkan Libfabric v1.22.0amzn4.0 atau lebih tinggi. Diuji dengan versi hingga Libfabric 2.1.0amzn4.0.<br>  • Menambahkan konfigurasi platform baru untuk mendukung penggunaan plugin OFI NCCL pada tipe instans Amazon EC2 p5.4xlarge<br>• Tingkatkan ke libfabric 2.1.0amzn5.0 | Juli 31, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.43.0 | • Tingkatkan ke libnccl-ofi 1.16.1<br>  • Mendukung NCCL v2.27.6-1 sambil mempertahankan kompatibilitas mundur dengan NCCL v2.17.1 dan yang lebih baru.<br>  • Membangun dengan platform-aws membutuhkan Libfabric v1.22.0amzn4.0 atau lebih tinggi. Diuji dengan versi hingga Libfabric 2.1.0amzn3.<br>  • Perbarui format kecepatan tautan PCI yang dilaporkan dalam file topologi agar sesuai dengan kernel 5.7+<br>  • Menambahkan SKIP_NICS_WITHOUT_ACCEL_AT_SAME_PCI_LEVEL untuk melewati nics libfabric yang tidak memiliki akselerator pada level pci yang sama<br>• Tingkatkan ke driver efa 2.17.2<br>• Tingkatkan ke efa-nv-peermem 1.2.2<br>• Tingkatkan ke libfabric 2.1.0amzn4.0<br>• Tingkatkan ke rdma-core 58.amzn0<br>  • Perbaiki penggunaan ganda indeks permintaan kerja<br>  • Tambahkan panjang WQE ke post_send tracepoint<br>• Aktifkan pengoptimalan untuk platform Graviton<br>• Menghentikan dukungan untuk Ubuntu 20.04 | Juli 25, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.42.0 | • Tingkatkan ke driver efa 2.15.3<br>• Tingkatkan ke efa-nv-peermem 1.2.1<br>• Tingkatkan ke rdma-core 57.amzn1<br>  • Perbaiki penggunaan ganda indeks permintaan kerja<br>• Tingkatkan ke libfabric 2.1.0amzn3.0<br>• Tingkatkan ke libnccl-ofi 1.15.0<br>  • Mendukung NCCL 2.26.6-1 sambil mempertahankan kompatibilitas mundur dengan (NCCL v2.17.1 dan yang lebih baru.<br>  • Membangun dengan platform-aws membutuhkan Libfabric v1.22.0amzn4.0 atau lebih tinggi. Diuji dengan versi hingga Libfabric 2.1.0amzn3.<br>  • Membangun dukungan sistem dan platform<br>  • Menambahkan dukungan P6-B200 platform Amazon EC2<br>  • Mengubah nama pustaka plugin default menjadi libnccl-net-ofi.so, dan secara default buat symlink dari libnccl-net-ofi.so ke libnccl-net.so untuk menjaga kompatibilitas ke belakang. Hal ini memungkinkan pengguna untuk mengatur NCCL_NET_PLUGIN=OFI untuk memaksa NCCL menggunakan plugin OFI untuk komunikasi. Menentukan --disable-nccl-net-symlink untuk dikonfigurasi akan melewati symlink, memungkinkan beberapa plugin diinstal dalam wadah yang sama.<br>  • Peningkatan penyetelan dan kinerja<br>  • Menambahkan dukungan tuner P6-B200 untuk AllReduce, AllGather, dan ReduceScatter wilayah untuk bitmask 0x0 dan 0x7 | Juni 6, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| | • Latensi default yang diperbarui untuk P5en dan P6-B200 platform berdasarkan hasil dan analisis empiris<br>• Perbarui untuk menggunakan API NCCL v10 dengan dukungan parameter TrafficClass untuk prioritas lalu lintas di masa mendatang<br>• Basis kode plugin yang dimigrasikan dari C ke C ++<br>• Menambahkan dukungan untuk pekerjaan di mana jumlah NIC per GPU berbeda di seluruh sistem. Lihat dokumenta si variabel lingkungan runtime OFI_NCCL_FORCE_NUM _RAILS untuk informasi lebih lanjut. | |
| 1.41.0 | • Tingkatkan ke rdma-core 57.amzn0 | 16 Mei 2025 |
| 1.40.0 | • Tingkatkan ke libfabric 2.1.0amzn2.0<br>• Tingkatkan ke rdma-core 56.0<br>• Tingkatkan ke efa-config 1.18.<br>• Menghentikan dukungan untuk Debian 10<br>• Tingkatkan ke libnccl-ofi 1.14.2<br>• Aktifkan dukungan CUDA di Libfabric di ARM untuk Ubuntu dan Amazon Linux 2023<br>• Tambahkan libnccl-ofi di ARM untuk Ubuntu dan Amazon Linux 2023 | 2 Mei 2025 |
| 1.39.0 | • Tingkatkan ke libfabric 2.1.0<br>• Tingkatkan ke Open MPI 5.0.6<br>• Tingkatkan ke libnccl-ofi 1.14.1<br>• Tingkatkan ke driver efa 2.15.0 | April 16, 2025 |
| 1.38.1 | • Perbarui ke `Libfabric 1.22.0amzn5.0` | Maret 3, 2025 |
| 1.38.0 | • Tambah `libnccl-ofi 1.13.2-1` | Januari 8, 2025 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.37.0 | • Tingkatkan ke `libfabric 1.22.0amzn4.0`<br>• Tingkatkan ke `rdma-core 54.amzn0` | November 18, 2024 |
| 1.36.0 | • Tambahkan dukungan untuk Debian 12<br>• Mendistribusikan `efa_test.sh` sebagai skrip utilitas di `/opt/amazon/efa/bin`<br>• Tingkatkan ke driver efa 2.13.0<br>• Tingkatkan ke `libfabric 1.22.0amzn3.0`<br>• Tingkatkan ke `rdma-core 54.0`<br>• Tingkatkan untuk membuka mpi 4.1.7 | 7 November 2024 |
| 1.35.0 | • Tingkatkan ke Open MPI 5.0.5<br>• Tingkatkan ke PRRTE 3.0.6 dan ganti nama paket RPM menjadi prrte-aws<br>• Ganti nama paket OpenPMix RPM menjadi pmix-aws<br>• Konfigurasikan opsi build untuk build dpkg dan RPM<br>• Tingkatkan ke `Libfabric 1.22.0amzn2.0`<br>• Tingkatkan ke driver efa 2.12.1 | Oktober 14, 2024 |
| 1.34.0 | • Menjatuhkan dukungan untuk CentOS 7 dan RHEL 7<br>• Menelan `Libfabric 1.22.0amzn1.0`<br>• Perbarui paket efa-config ke 1.17 | Agustus 6, 2024 |
| 1.33.0 | • Tingkatkan ke driver efa 2.10.0<br>• Tingkatkan ke `rdma-core 52.0`<br>• Tingkatkan ke PRRTE 3.0.5<br>• Tingkatkan ke Open MPI 5.0.3<br>• Tambahkan dukungan untuk Ubuntu 24.04 LTS<br>• Hapus dokumentasi OpenPMix, PRRTE dan Open MPI HTML. Pengguna harus merujuk ke situs web resmi.<br>• Hapus dukungan untuk openSUSE 15.4 atau yang lebih lama | Juni 20, 2024 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.32.0 | • Tingkatkan ke driver efa 2.8.0<br>• Tingkatkan ke driver efa-nv-peermem 1.1.1<br>• Tingkatkan paket efa-config ke 1.16<br>• Tingkatkan paket efa-profile ke 1.7<br>• Tingkatkan ke `rdma-core 50.0`<br>• Tingkatkan ke `libfabric 1.21.0amzn1.0`<br>• Tingkatkan `efa_test.sh` dengan pemilihan `fi_pingpong` port<br>• Instal rdma-core yang lebih baru dari sistem bila diperlukan | April 18, 2024 |
| 1.31.0 | • Tingkatkan ke OpenPMix 4.2.8<br>• Tingkatkan ke PRRTE 3.0.3<br>• Tingkatkan ke Open MPI 5.0.2, dan konversi komponen MCA ke DSO.<br>• Tingkatkan ke `Libfabric 1.20.1amzn1.0` | Maret 7, 2024 |
| 1.30.0 | • Sertakan OpenPMix 4.2.7, diinstal pada secara default `/opt/amazon/pmix`<br>• Sertakan PRRTE 3.0.2, diinstal pada secara default `/opt/amazon/prrte`<br>• Sertakan Open MPI 5.0.0, diinstal pada secara `/opt/amazon/openmpi5` default<br>• Perbarui efa-profile menjadi 1.6 | Desember 2023 |
| 1.29.1 | • Menelan `libfabric 1.19.0amzn4.0` | Desember 2023 |
| 1.29.0 | • Mengambil driver kernel efa 2.6.0<br>• Menelan `libfabric 1.19.0amzn3.0` | November 2023 |
| 1.28.0 | • Tingkatkan Open MPI ke 4.1.6<br>• Menelan `libfabric 1.19.0amzn2.0` | Oktober 2023 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.27.0 | • Tambahkan `Libfabric 1.19.0amzn1.0` ke penginstal<br>• Aktifkan atomik bawaan GCC untuk Open MPI 4 | September 2023 |
| 1.26.1 | • `efa_test.sh` : perbaiki bug percobaan ulang tak terbatas | September 2023 |
| 1.26.0 | • Tambahkan dukungan untuk RHEL 9<br>• Perbaiki build debug pada beberapa distribusi<br>• Menelan `libfabric 1.18.2amzn1.0`<br>• `efa_test.sh` : Tambahkan coba lagi untuk `fi_pingpong` | September 2023 |
| 1.25.1 | • Menelan `libfabric 1.18.1amzn1.0`<br>• Perbarui batas waktu di skrip penginstal EFA | September 2023 |
| 1.25.0 | • Tambahkan dukungan untuk Amazon Linux 2023<br>• Perbaikan bug dalam tes pasca-instalasi | Juli 2023 |
| 1.24.1 | • Tingkatkan libfabric ke 1.18.1<br>• Tingkatkan driver efa ke 2.5.0 | Juli 2023 |
| 1.24.0 | • Menelan `rdma-core 46.0`<br>• Ingest driver efa 2.4.1<br>• Mendukung Debian 11 | Juni 2023 |
| 1.23.1 | • Menelan `libfabric 1.18.0amzn2.0` | Juni 2023 |
| 1.23.0 | • Tambahkan dukungan untuk Debian 10<br>• Menjatuhkan dukungan untuk Ubuntu 18.04 LTS<br>• Tingkatkan paket efa-config ke 1.14<br>• Menelan `libfabric 1.18.0amzn1.0` | Mei 2023 |
| 122.1 | • Tingkatkan libfabric ke 1.17.1 | Maret 2023 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.22.0 | • Tingkatkan Open MPI ke 4.1.5<br>• Tingkatkan libfabric ke 1.17.0<br>• Tingkatkan paket efa-config ke 1.13 | Februari 2023 |
| 1.21.0 | • Tambahkan dukungan untuk Rocky Linux 9 OS<br>• Ingest driver efa 2.1.1<br>• Menelan `libfabric 1.16.1amzn3.0`<br>• Tingkatkan paket efa-config ke 1.12 | Desember 2022 |
| 1.20.0 | • Tambahkan dukungan untuk Rocky Linux 8 OS.<br>• Ingest driver efa 2.1.0.<br>• Menelan`rdma-core 43.0`.<br>• Menelan `libfabric 1.16.1amzn1.0` | November 2022 |
| 1.19.0 | • Menelan `libfabric 1.16.0`<br>• Membangun Open MPI dengan `--enable-orterun-prefix-by-default` | Oktober 2022 |
| 1.18.0 | • Tambahkan dukungan untuk Ubuntu22.04 | Agustus 2022 |
| 1.17.3 | • Perbarui libfabric ke`1.16.0~amzn4.0` . Ini ~ menunjukkan itu adalah versi pra-rilis dari libfabric 1.16.0.<br>• Perpanjang batas waktu tes pingpong pasca instalasi hingga 20 detik | Agustus 2022 |
| 1.17.2 | • Perbarui libfabric ke`1.16.0~amzn3.0` . Ini ~ menunjukkan itu adalah versi pra-rilis dari libfabric 1.16.0. | Juli 2022 |
| 1.17.1 | • Perbarui libfabric ke`1.16.0~amzn2.0` . Ini ~ menunjukkan itu adalah versi pra-rilis dari libfabric 1.16.0.<br>• Nonaktifkan penyedia net eksperimental saat membangun libfabric | Juli 2022 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.17.0 | • Perbarui `rdma-core` ke v41.0<br>• Perbarui Open MPI ke 4.1.4<br>• Perbarui libfabric ke`1.16.0~amzn1.0` . Ini ~ menunjukkan itu adalah versi pra-rilis dari libfabric 1.16.0. | Juli 2022 |
| 1.16.0 | • Perbarui libfabric menjadi`1.15.1amzn1.0` , berisi perubahan nama perpustakaan neuron<br>• Tingkatkan efa-config ke 1.10<br>• Kecualikan penyedia opx dan rxd di build libfabric | Juni 2022 |
| 1.15.2 | • Perbarui libfabric ke 1.14.1 | Mei 2022 |
| 1.15.1 | • Perbarui libfabric ke `1.14.0amzn1.0` | Maret 2022 |
| 1.15.0 | • Perbaiki bug yang menyebabkan instalasi gagal pada Open SuSE 15.3<br>• Menjatuhkan dukungan Open SuSE 15.2 (karena Open SuSE 15.2 mencapai akhir masa pakai)<br>• Menjatuhkan Dukungan CentOS 8 (saat CentOS 8 mencapai akhir masa pakai)<br>• Perbarui libfabric menjadi 1.14.0<br>• Perbarui driver kernel efa ke 1.16.0<br>• Perbarui `rdma-core` ke v39.0<br>• Perbarui Open MPI ke versi 4.1.2. | Februari 2022 |
| 1.14.1 | • Perbarui libfabric ke`1.13.2amzn1.0` . | Oktober 2021 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.14.0 | • Mengambil driver kernel efa 1.14.2.<br>• J `-g, --enable-gdr` adikan `efa_installer.sh` sebagai opsi tanpa operasi karena driver kernel efa terbaru mengaktifkan dukungan GDR secara default.<br>• Menelan`rdma-core v37.0`.<br>• Menelan`libfabric 1.13.2`.<br>• Tambahkan daftar paket dan bandingkan dengan RPM/DEB daftar selama instalasi untuk mencegah instalasi paket yang tidak dikenal.<br>• Tambahkan tidur dalam skrip penginstal untuk menunggu aturan udev diterapkan setelah pemuatan ulang driver EFA. | Oktober 2021 |
| 1.13.0 | • Perbarui `rdma-core` ke v35.0.<br>• Perbarui `libfabric` ke v1.13.0amzn1.0.<br>• Tambahkan dukungan EFA untuk CentOS/RHEL 8 pada platform Gravition2.<br>• Tambahkan logika perbandingan versi dalam skrip penginsta l untuk melewati instalasi paket lokal ketika ada satu yang diinstal dalam sistem dengan versi yang lebih tinggi. | Agustus 2021 |
| 1.12.3 | • Perbarui modul kernel EFA ke 1.13.0.<br>• Perbarui paket efa-config ke versi 1.9. Tingkatkan perhitungan reservasi halaman besar untuk menangani ukuran halaman besar default besar. | Juli 2021 |
| 1.12.2 | • Perbarui modul kernel EFA ke 1.12.3.<br>• Bangun paket debian MPI terbuka dengan `--with-li bevent=external` dan`--with-hwloc=external` .<br>• Bump Open MPI rpm build ID ke 2 untuk memperbaiki masalah kompatibilitas mundur HWLOC pada CentOS 8.<br>• Hapus instalasi paket kernel-devel dan kernel-source pada SLES15SP2 dan openSUSE 15.2. | Juni 2021 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.12.1 | • Perbarui Libfabric ke versi 1.11.2amzon1.1.<br>• Perbarui modul kernel EFA ke versi 1.12.1. | Mei 2021 |
| 1.12.0 | • Perbarui Open MPI ke versi 4.1.1.<br>• Perbarui Libfabric ke versi 1.11.2amzn1.0.<br>• Bangun rdma-core untuk Amazon Linux 2 menggunakan konfigurasi pengemasan yang sama dengan RDMA-core AL2.<br>• Jangan memaksakan `-Wl,--enable-new-dtags` saat membangun RPM MPI Terbuka.<br>• Bangun Open MPI dengan pustaka sistem untuk hwloc dan libevent.<br>• Perbarui modul kernel EFA ke versi 1.12.0<br>• Perbarui paket efa-config ke versi 1.8. Tingkatkan perhitungan reservasi halaman besar untuk instans berumur panjang.<br>• Perbarui paket efa-profile ke versi 1.5. Hapus file penyetela n kolektif mpi terbuka yang berfungsi sebagai solusi untuk memperbaiki Open MPI 4.1.0 hang di P4d.<br>• Perbarui `rdma-core` ke v32.1.<br>• Menjatuhkan dukungan untuk Amazon Linux 1 dan Ubuntu 16.04. | Mei 2021 |
| 1.11.2 | • Perbaiki Open MPI hang saat menggunakan Open MPI di P4d dengan mengubah algoritma default yang digunakan untuk mengimplementasikan MPI_BARRIER melalui file konfigurasi.<br>• Nonaktifkan penggunaan atomik bawaan di Open MPI di ARM melalui `--disable-builtin-atomics` untuk mengatasi masalah kompiler. | Februari 2021 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.11.1 | • Perbarui Open MPI ke versi 4.1.0.<br>• Perbarui paket efa-config ke versi 1.7. Tingkatkan perhitungan jumlah reservasi halaman besar.<br>• Perbarui paket efa-profile ke versi 1.3. Menghapus file keputusan kolektif yang tidak diperlukan sekarang setelah Open MPI 4.1.0 digunakan. | Desember 2020 |
| 1.1.0 | • Tambahkan dukungan untuk platform Gravition2.<br>• Perbarui `rdma-core` ke versi 31.2amzn.<br>• Perbarui Libfabric ke versi 1.11.1amzn1.0.<br>• Perbarui efa-config ke versi 1.6.<br>• Perbarui efa-profile ke versi 1.2. | Desember 2020 |
| 1.10.1 | • Tambahkan dukungan untuk CentOS/RHEL 8.<br>• Tambahkan dukungan untuk Ubuntu 20.04.<br>• Tambahkan dukungan untuk SUSE Linux Enterprise 15. | November 2020 |
| 1.10.0 | • Tambahkan dukungan RDMA GPUdirect untuk platform P4d. Gunakan opsi `--enable-gdr` penginstal untuk menginstal modul GDR-aware kernel dan ruang pengguna.<br>• Perbarui modul kernel EFA ke versi 1.10.2.<br>• Perbarui `rdma-core` ke versi 31.amzn0.<br>• Perbarui Libfabric ke versi 1.11.1.<br>• Perbarui Open MPI ke versi 4.0.5.<br>• Perbarui efa-config ke versi 1.5.<br>• Perbarui efa-profile ke versi 1.1. Termasuk file keputusan kolektif Open MPI yang ditingkatkan. | Oktober 2020 |
| 1.9.5 | • Perbarui efa-config ke versi 1.4. Memperbaiki bug di file keputusan kolektif Open MPI. | September 2020 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.9.4 | • Perbarui Open MPI ke versi 4.0.3.<br>• Perbarui Libfabric ke versi 1.10.1amazon1.1.<br>• Perbarui `rdma-core` ke versi 28.amzn0. | Juli 2020 |
| 1.9.3 | • Perbarui modul kernel EFA ke versi 1.6.0.<br>• Perbarui `rdma-core` ke versi 28.amzn0.<br>• Perbarui Libfabric ke versi 1.10.1amzn1.1.<br>• Perbarui efa-config ke versi 1.3. Menambahkan file tuning kolektif untuk Open MPI.<br>• Lewati instalasi dkms jika sudah diinstal.<br>• Perbaiki mode `--skip-kmod` instalasi agar benar-benar berfungsi. | Juni 2020 |
| 1.8.4 | • Pindahkan file konfigurasi ke paket efa-config dan efa-profile sehingga dilacak oleh manajer paket sistem operasi.<br>• Perbarui Open MPI ke versi 4.0.3. | April 2020 |
| 1.8.3 | • Perbarui modul kernel EFA ke versi 1.5.1.<br>• DKMS terdistribusi pada beberapa platform daripada mengandalkan repositori EPEL untuk keandalan instalasi tambahan.<br>• Pada RHEL 7, instal RPM yang dibangun di CentOS 7 alih-alih RPM yang dibangun di Amazon Linux 2. | Februari 2020 |
| 1.8.2 | • Kembali `rdma-core` ke versi 25 karena ketidakcocokan dalam penamaan perangkat antara modul kernel dan. `rdma-core` | Januari 2020 |
| 1.8.1 | • Perbarui Libfabric ke versi 1.9.0amzn1.1. | Januari 2020 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.8.0 | • Perbarui `rdma-core` ke versi 27.0.<br>• Perbarui modul kernel EFA ke versi 1.5.0.<br>• Perbarui Libfabric ke versi 1.9.0amzn1.0.<br>• Tambahkan opsi `--minimal` untuk menginstal hanya modul kernel EFA dan`rdma-core` . | Desember 2019 |
| 1.7.1 | • Perbarui Libfabric ke versi 1.8.1amzn1.3. | Desember 2019 |
| 1.7.0 | • Tambahkan file modul Libfabric.<br>• Perbarui Libfabric ke versi 1.8.1amzn1.1. | November 2019 |
| 1.6.2 | • Perbarui Open MPI ke versi 1.6.2. | Oktober 2019 |
| 1.6.1 | • Perbarui Libfabric ke versi 1.8.1amzn1.0.<br>• Perbarui Open MPI ke versi 4.0.1.<br>• Perbarui `rdma-core` ke versi 26.0. | Oktober 2019 |
| 1.5.4 | • Perbarui modul kernel EFA ke versi 1.4.1. | September 2019 |
| 1.5.3 | • Perbarui modul kernel EFA ke versi 1.3.1.<br>• Hindari menginstal `kernel-devel` atau `linux-headers` paket kecuali menginstal driver kernel. | September 2019 |
| 1.5.1 | • Hanya mengkonfigurasi halaman besar saat perangkat EFA hadir. | Agustus 2019 |
| 1.5.0 | • Perbarui Libfabric ke versi 1.8.0amzn1.1.<br>• Perbarui `rmda-core` ke versi 25.0. | Agustus 2019 |

| Versi | Perubahan | Tanggal rilis |
| --- | --- | --- |
| 1.4.1 | • Tambahkan jalur pustaka Libfabric dan Open MPI (`/opt/amazon/efa/lib64` dan `/opt/amazon/efa/openmpi/lib`) `/etc/ld.so.conf.d/efa.conf` untuk memastikan pustaka bersama Open MPI dan Libfabric ditempatkan dengan benar. | Juli 2019 |
| 1.4.0 | • Perbarui modul kernel EFA ke versi 1.3.0.<br>• Perbarui Libfabric ke versi 1.8.0amzn1.0<br>• Rilis pertama untuk mendukung Intel MPI 2019 Update 4. | Juli 2019 |

# Topologi Amazon EC2

Topologi Amazon EC2 memberikan tampilan hierarkis tentang kedekatan relatif kapasitas komputasi Anda. Anda dapat menggunakan informasi ini untuk mengelola komputasi kinerja tinggi (HPC), pembelajaran mesin (ML), dan infrastruktur komputasi AI generatif dalam skala besar.

API yang tersedia

Amazon EC2 menyediakan dua API untuk memahami topologi EC2 Anda:

- DescribeInstanceTopology
  - Menunjukkan lokasi instans yang sedang berjalan relatif satu sama lain dalam hierarki jaringan.
  - Membantu mengoptimalkan tempat menjalankan beban kerja pada instans yang ada.
- DescribeCapacityReservationTopology
  - Menunjukkan lokasi kapasitas cadangan Anda relatif terhadap satu sama lain dalam hierarki jaringan sebelum Anda meluncurkan instans.
  - Membantu perencanaan kapasitas dengan memberi tahu Anda penempatan kapasitas cadangan sebelum meluncurkan instans.

Manfaat utama

Topologi EC2 memberikan manfaat utama berikut:

- Manajemen kapasitas — Mengoptimalkan pemanfaatan sumber daya.

- Penjadwalan pekerjaan — Buat keputusan berdasarkan informasi tentang penempatan beban kerja.
- Peringkat simpul — Memahami kedekatan relatif untuk pengoptimalan kinerja pada instans yang digabungkan erat.

Pertimbangan-pertimbangan

- Tampilan topologi hanya tersedia untuk:
  - Contoh di `running` negara bagian
  - Reservasi Kapasitas di `active` negara `pending` bagian atau
- Setiap tampilan topologi unik per. Akun AWS
- Konsol Manajemen AWS Tidak mendukung tampilan topologi.
- Meskipun informasi topologi membantu Anda memahami penempatan instance, Anda tidak dapat menggunakannya untuk meluncurkan instans baru secara fisik dekat dengan instance yang sudah ada. Untuk memengaruhi penempatan instans, Anda dapat membuat Reservasi Kapasitas di grup penempatan cluster.

Harga

Tidak ada biaya tambahan untuk menggambarkan topologi EC2 Anda.

Daftar Isi

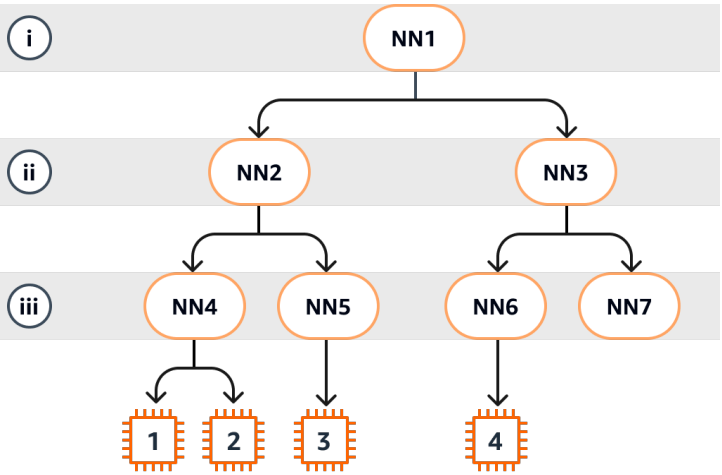

- Cara kerja topologi Amazon EC2
- Prasyarat untuk topologi Amazon EC2
- Contoh topologi instans Amazon EC2


## Cara kerja topologi Amazon EC2

AWS Jaringan diatur dalam hierarki lapisan. Instans EC2 terhubung ke jaringan pada atau di bawah lapisan ketiga, tergantung pada jenis instans. Topologi instance dijelaskan oleh satu set node, dengan satu node di setiap lapisan jaringan. Node yang disetel dalam respon DescribeCapacityReservationTopology API DescribeInstanceTopology or menyediakan tampilan top-down dari hierarki jaringan, dengan node bawah terhubung ke instance.

> **Note**
>
> Beberapa jenis instance terdiri dari 4 node jaringan dalam satu set node yang mewakili 4 lapisan dalam jaringan, sementara yang lain terdiri dari 3 node jaringan yang mewakili 3 lapisan dalam jaringan. Untuk jenis instans yang didukung, lihatTipe instans.
> Bergantung pada jenis Reservasi Kapasitas, Anda mungkin hanya melihat 1, 2, atau 3 node jaringan.

Diagram berikut memberikan representasi visual yang dapat Anda gunakan untuk memahami topologi EC2. Node jaringan diidentifikasi sebagai NN1 - N N 7. Angka i, ii, dan iii mengidentifikasi lapisan jaringan. Angka 1, 2, 3, dan 4 mengidentifikasi instance EC2. Instans terhubung ke node di lapisan bawah, diidentifikasi oleh iii dalam diagram berikut. Lebih dari satu instans dapat terhubung ke simpul yang sama.




Dalam contoh ini:

- Instance 1 terhubung ke node jaringan 4 (NN4) di lapisan iii. NN4 terhubung ke simpul jaringan 2 (NN2) di lapisan ii, dan NN2 terhubung ke simpul jaringan 1 (NN1) di lapisan i, yang merupakan bagian atas hierarki jaringan dalam contoh ini. Kumpulan simpul jaringan terdiri dari NN1, NN2, dan NN4, diekspresikan secara hierarkis dari lapisan atas ke lapisan bawah.
- Instans 2 juga terhubung ke simpul jaringan 4 (NN4). Instans 1 dan instans 2 berbagi rangkaian simpul jaringan yang sama: NN1, NN2, dan NN4.
- Instans 3 terhubung ke simpul jaringan 5 (NN5). NN5 terhubung ke NN2, dan NN2 terhubung ke NN1. Simpul jaringan yang ditetapkan untuk instans 3 adalah NN1, NN2, dan NN5.
- Instans 4 terhubung ke simpul jaringan 6 (NN6). Set simpul jaringannya adalah NN1, NN3, dan NN6.

Ketika mempertimbangkan kedekatan instans 1, 2, dan 3, instans 1 dan 2 lebih dekat satu sama lain karena mereka terhubung ke simpul jaringan yang sama (NN4), sedangkan instans 3 lebih jauh karena terhubung ke simpul jaringan yang berbeda (NN5).

Ketika mempertimbangkan kedekatan semua instans dalam diagram ini, instans 1, 2, dan 3 lebih dekat satu sama lain daripada instans 4 karena mereka berbagi NN2 dalam rangkaian simpul jaringan mereka.

Sebagai aturan umum, jika simpul jaringan yang terhubung ke dua instans adalah sama, instans ini secara fisik dekat satu sama lain, seperti halnya dengan instans 1 dan 2. Selanjutnya, makin sedikit jumlah lompatan antara simpul jaringan, makin dekat instans satu sama lain. Misalnya, instans 1 dan 3 memiliki lebih sedikit lompatan ke simpul jaringan umum (NN2) daripada yang mereka miliki ke simpul jaringan (NN1) yang mereka miliki bersama dengan instans 4, dan karena itu lebih dekat satu sama lain daripada instans 4.

Tidak ada instans yang berjalan di bawah simpul jaringan 7 (NN7) dalam contoh ini, dan oleh karena itu output API tidak akan menyertakan NN7.

## Bagaimana menafsirkan DescribeInstanceTopology output

Anda dapat menjelaskan topologi instance menggunakan DescribeInstanceTopology API. Output memberikan tampilan hierarkis topologi jaringan yang mendasarinya untuk sebuah instance.

Contoh output berikut sesuai dengan informasi topologi jaringan dari empat instans dalam diagram sebelumnya. Komentar disertakan dalam contoh output untuk keperluan contoh ini.

Informasi berikut dalam output penting untuk dicatat:

- `NetworkNodes`menjelaskan kumpulan node jaringan dari satu instance.
- Dalam setiap set simpul jaringan, simpul jaringan terdaftar dalam urutan hierarkis dari atas ke bawah.
- Simpul jaringan yang terhubung ke instans adalah simpul jaringan terakhir dalam daftar (lapisan bawah).
- Untuk mengetahui instans mana yang dekat satu sama lain, pertama-tama temukan simpul jaringan umum di lapisan bawah. Jika tidak ada simpul jaringan umum di lapisan bawah, maka temukan simpul jaringan umum di lapisan atas.

Dalam contoh output berikut, `i-1111111111example` dan `i-2222222222example` terletak paling dekat satu sama lain dibandingkan dengan instans lain dalam contoh ini karena mereka memiliki simpul jaringan yang sama `nn-4444444444example` di lapisan bawah.

> **Note**
>
> Respons berisi 3 atau lebih node jaringan. Untuk informasi tentang jumlah node jaringan dalam respons untuk setiap jenis instans yang didukung, lihatTipe instans.

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",  //Corresponds to instance 1
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",           //Corresponds to NN1 in layer i
                "nn-2222222222example",           //Corresponds to NN2 in layer ii
                "nn-4444444444example"            //Corresponds to NN4 in layer iii -
 bottom layer, connected to the instance
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",  //Corresponds to instance 2
            "InstanceType": "p4d.24xlarge",
            "NetworkNodes": [
                "nn-1111111111example",           //Corresponds to NN1 - layer i
                "nn-2222222222example",           //Corresponds to NN2 - layer ii
                "nn-4444444444example"            //Corresponds to NN4 - layer iii -
 connected to instance
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-3333333333example",  //Corresponds to instance 3
            "InstanceType": "trn1.32xlarge",
```

```
            "NetworkNodes": [
                "nn-1111111111example",           //Corresponds to NN1 - layer i
                "nn-2222222222example",           //Corresponds to NN2 - layer ii
                "nn-5555555555example"            //Corresponds to NN5 - layer iii -
 connected to instance
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-444444444example",  //Corresponds to instance 4
            "InstanceType": "trn1.2xlarge",
            "NetworkNodes": [
                "nn-1111111111example",          //Corresponds to NN1 - layer i
                "nn-3333333333example",          //Corresponds to NN3 - layer ii
                "nn-6666666666example"           //Corresponds to NN6 - layer iii -
 connected to instance
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

## Bagaimana menafsirkan DescribeCapacityReservationTopology output

Anda dapat menjelaskan topologi Reservasi Kapasitas menggunakan DescribeCapacityReservationTopology API. Output memberikan tampilan hierarkis topologi jaringan yang mendasarinya untuk kapasitas yang dicadangkan.

Contoh output berikut sesuai dengan informasi topologi jaringan dalam diagram sebelumnya. Komentar disertakan dalam contoh output untuk keperluan contoh ini.

Informasi berikut dalam output penting untuk dicatat:

- `NetworkNodes`menjelaskan kumpulan node jaringan dari satu Reservasi Kapasitas.
- Dalam setiap set simpul jaringan, simpul jaringan terdaftar dalam urutan hierarkis dari atas ke bawah.

- Node jaringan yang terhubung ke Reservasi Kapasitas adalah node jaringan terakhir dalam daftar (lapisan bawah).
- Untuk mengetahui apakah Reservasi Kapasitas akan dekat satu sama lain, pertama-tama temukan node jaringan umum di lapisan bawah dalam output. Jika tidak ada simpul jaringan umum di lapisan bawah, maka temukan simpul jaringan umum di lapisan atas.

Dalam contoh berikut output, `cr-1111111111example` terletak di `nn-2222222222example` dan `cr-2222222222example` terletak di`nn-3333333333example`. Karena Reservasi Kapasitas berada di node jaringan yang berbeda di`layer ii`, komunikasi dari instans dalam satu Reservasi Kapasitas ke instans di Reservasi Kapasitas lainnya tidak akan efisien.

> **Note**
>
> Respons berisi 1, 2, atau 3 node jaringan tergantung pada jenis Reservasi Kapasitas.

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-1111111111example",
            "CapacityBlockId": "null",
            "State": "active",
            "InstanceType": "p4d.24xlarge",
            "NetworkNodes": [
                "nn-1111111111example",      //Corresponds to NN1 - layer i
                "nn-2222222222example"       //Corresponds to NN2 - layer ii
                // Visibility of additional nodes requires an instance launch and
                // the DescribeInstanceTopology API
            ],
            "AvailabilityZone": "us-west-2a"
        },
        {
            "CapacityReservationId": "cr-2222222222example",
            "CapacityBlockId": "null",
            "State": "active",
            "InstanceType": "trn1.2xlarge",
            "NetworkNodes": [
                "nn-1111111111example",      //Corresponds to NN1 - layer i
                "nn-3333333333example"       //Corresponds to NN3 - layer ii
                // Visibility of additional nodes requires an instance launch and
```

```
                // the DescribeInstanceTopology API
            ],
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

## Perbedaan antara DescribeInstanceTopology dan DescribeCapacityReservationTopology

Tabel berikut membandingkan perbedaan utama antara DescribeInstanceTopology dan DescribeCapacityReservationTopology API:

| Titik perbandingan | DescribeInstanceTopology | DescribeCapacityReservation Topology |
|---|---|---|
| Fase penggunaan | Post-launch (modus eksekusi) | Pre-launch (mode perencana an dan manajemen) |
| Tujuan utama | Mengoptimalkan beban kerja pada instans yang sedang berjalan | Perencanaan kapasitas dan manajemen Reservasi Kapasitas (menggabungkan, membagi, menetapkan) sebelum peluncuran instans |
| Jumlah node jaringan | Menampilkan semua node untuk instance yang sedang berjalan. Jika instance berada dalam Reservasi Kapasitas , node pertama akan cocok dengan topologi Reservasi Kapasitas yang sesuai, diikuti oleh node tambahan untuk terhubung ke instance. | Menampilkan kumpulan sebagian node, yang bervariasi berdasarkan status Reservasi Kapasitas (`pendingatauactive`) dan jenis. * |

| Titik perbandingan | DescribeInstanceTopology | DescribeCapacityReservation Topology |
| --- | --- | --- |
| Status | Instansi harus dalam `running` negara | Reservasi Kapasitas harus berada di `pending` atau `active` negara |
| Kasus penggunaan | • Optimalisasi beban kerja<br>• Penyempurnaan performa<br>• Analisis topologi runtime | • Perencanaan kapasitas<br>• Manajemen reservasi kapasitas (merge/split/ menetapkan)<br>• Pre-launch penilaian topologi |

* Untuk Blok Kapasitas untuk Ultraservers, set node jaringan sama ketika menjelaskan topologi untuk Reservasi `active` Kapasitas atau instans yang sedang berjalan.

## Prasyarat untuk topologi Amazon EC2

Untuk menjelaskan topologi Amazon EC2 Anda, pastikan instans dan Reservasi Kapasitas memenuhi prasyarat berikut.

Prasyarat untuk:

- Wilayah AWS
- Tipe instans
- Status
- Izin IAM

### Wilayah AWS

Didukung Wilayah AWS:

- AS Timur (Virginia Utara), AS Timur (Ohio), AS Barat (California Utara), AS Barat (Oregon)
- Africa (Cape Town)

- Asia Pasifik (Jakarta), Asia Pasifik (Hong Kong), Asia Pasifik (Hyderabad), Asia Pasifik (Melbourne), Asia Pasifik (Mumbai), Asia Pasifik (Osaka), Asia Pasifik (Seoul), Asia Pasifik (Singapura), Asia Pasifik (Sydney), Asia Pasifik (Tokyo)
- Kanada (Pusat)
- Eropa (Frankfurt), Eropa (Irlandia), Eropa (London), Eropa (Paris), Eropa (Spanyol), Eropa (Stockholm), Eropa (Zurich)
- Israel (Tel Aviv)
- Timur Tengah (Bahrain), Timur Tengah (UEA)
- Amerika Selatan (Sao Paulo)
- AWS GovCloud (US-West)
- AWS European Sovereign Cloud (Germany)

DescribeCapacityReservationTopology API tidak didukung di Israel (Tel Aviv) dan AWS GovCloud (US-West).

## Tipe instans

Tipe instans yang didukung:

- Meng  embalikan 3* node jaringan  dalam respons:
  - `g6e.xlarge` | `g6e.2xlarge` | `g6e.4xlarge` | `g6e.8xlarge` | `g6e.12xlarge` | `g6e.16xlarge` | `g6e.24xlarge` | `g6e.48xlarge` | `g7e.2xlarge` | `g7e.4xlarge` | `g7e.8xlarge` | `g7e.12xlarge` | `g7e.24xlarge` | `g7e.48xlarge`
  - `hpc6a.48xlarge` | `hpc6id.32xlarge` | `hpc7g.4xlarge` | `hpc7g.8xlarge` | `hpc7g.16xlarge` | `hpc7a.12xlarge` | `hpc7a.24xlarge` | `hpc7a.48xlarge` | `hpc7a.96xlarge` | `hpc8a.96xlarge`
  - `p3dn.24xlarge` | `p4d.24xlarge` | `p4de.24xlarge` | `p5.48xlarge` | `p5e.48xlarge` | `p5en.48xlarge` | `p6e-gb200.36xlarge`
  - `trn1.2xlarge` | `trn1.32xlarge` | `trn1n.32xlarge` | `trn2.48xlarge` | `trn2u.48xlarge`
- Meng  embalikan 4* node jaringan  dalam respons:
  - `p6-b200.48xlarge` | `p6-b300.48xlarge`

* Jumlah node jaringan yang dikembalikan hanya berlaku saat menggunakan DescribeInstanceTopology API. Untuk DescribeCapacityReservationTopology API, jumlah node jaringan yang dikembalikan akan bervariasi tergantung pada jenis dan status Reservasi Kapasitas.

Tipe instans yang tersedia berbeda-beda menurut Wilayah. Untuk informasi selengkapnya, lihat Jenis instans Amazon EC2 menurut Wilayah.

### Status

- Untuk `DescribeInstanceTopology` - Contoh harus di `running` negara bagian.
- Untuk `DescribeCapacityReservationTopology` - Reservasi Kapasitas harus di `active` negara `pending` bagian atau.

Anda tidak bisa mendapatkan informasi topologi untuk instans atau Reservasi Kapasitas di negara bagian lain mana pun.

### Izin IAM

Identitas IAM Anda (pengguna, grup pengguna, atau peran) memerlukan izin berikut:

- `ec2:DescribeInstanceTopology`
- `ec2:DescribeCapacityReservationTopology`

## Contoh topologi instans Amazon EC2

Anda dapat menggunakan describe-instance-topology perintah untuk menjelaskan topologi untuk instans EC2 Anda. Dan Anda dapat menggunakan describe-capacity-reservation-topology perintah untuk menggambarkan topologi Reservasi Kapasitas Anda.

Saat Anda menggunakan `describe-capacity-reservation-topology` perintah `describe-instance-topology` or tanpa parameter atau filter, respons mencakup semua instans atau Reservasi Kapasitas Anda (tergantung pada perintah yang digunakan) yang cocok dengan jenis instans yang didukung untuk perintah ini di Wilayah yang ditentukan. Anda dapat menentukan Wilayah dengan menyertakan parameter `--region`, atau dengan menetapkan Wilayah default. Untuk informasi selengkapnya tentang mengatur Wilayah default, lihat Pilih Wilayah untuk sumber daya Amazon EC2 Anda.

Anda dapat menyertakan parameter untuk mengembalikan instans atau Reservasi Kapasitas yang cocok dengan instans tertentu atau ID Reservasi Kapasitas atau nama grup penempatan. Anda

juga dapat menyertakan filter untuk mengembalikan instans atau Reservasi Kapasitas yang sesuai dengan jenis instans atau keluarga instans tertentu, atau instans atau Reservasi Kapasitas di Zona Ketersediaan atau Zona Lokal yang ditentukan. Anda dapat menyertakan satu parameter atau filter, atau kombinasi parameter dan filter.

Output dipaginasi, dengan hingga 20 instans atau Reservasi Kapasitas per halaman secara default. Anda dapat menentukan hingga 100 instans atau Reservasi Kapasitas per halaman menggunakan `--max-results` parameter.

Untuk informasi selengkapnya, lihat describe-instance-topology dan describe-reservation-topology-topology.

Izin yang diperlukan

Izin berikut diperlukan:

- `ec2:DescribeInstanceTopology`— Untuk menggambarkan topologi contoh
- `ec2:DescribeCapacityReservationTopology`— Untuk menjelaskan topologi Reservasi Kapasitas

Contoh

- Contoh 1: DescribeInstanceTopology - ID Instance
- Contoh 2: DescribeInstanceTopology - Parameter nama grup penempatan
- Contoh 3: DescribeInstanceTopology - Filter jenis instans
  - Contoh 3a — Filter kecocokan tepat untuk jenis instance tertentu
  - Contoh 3b — Filter kartu liar untuk keluarga instance
  - Contoh 3c — Keluarga instance gabungan dan filter kecocokan tepat
- Contoh 4: DescribeInstanceTopology - Filter ID Zona
  - Contoh 4a — Filter Zona Ketersediaan
  - Contoh 4b — Filter Zona Lokal
  - Contoh 4c — Filter Zona Ketersediaan Gabungan dan Zona Lokal
- Contoh 5: DescribeInstanceTopology - Jenis instans dan filter ID zona
- Contoh 6: DescribeCapacityReservationTopology - ID Reservasi Kapasitas
- Contoh 7: DescribeCapacityReservationTopology - Filter jenis instans

## Contoh 1: DescribeInstanceTopology - ID Instance

AWS CLI

Untuk menggambarkan topologi contoh tertentu

Gunakan describe-instance-topology perintah dengan `--instance-ids` parameter. Output hanya mencakup instance yang cocok dengan ID instance yang ditentukan.

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --instance-ids i-1111111111example i-2222222222example
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "GroupName": "HPC-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
```

```
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menggambarkan topologi contoh tertentu

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -InstanceId i-1111111111example, i-2222222222example
```

## Contoh 2: DescribeInstanceTopology - Parameter nama grup penempatan

AWS CLI

Untuk menggambarkan topologi instance dalam kelompok penempatan tertentu

Gunakan describe-instance-topology perintah dengan `group-names` parameter. Output hanya mencakup instance yang ada di salah satu grup penempatan yang ditentukan.

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --group-names ML-group HPC-group
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
```

```
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "GroupName": "HPC-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menggambarkan topologi instance dalam kelompok penempatan tertentu

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -GroupName ML-group, HPC-group
```

## Contoh 3: DescribeInstanceTopology - Filter jenis instans

Anda dapat memfilter berdasarkan tipe instans tertentu (sama persis) atau memfilter menurut keluarga instans (menggunakan wildcard). Anda juga dapat menggabungkan filter tipe instans tertentu dan filter keluarga instans.

Contoh 3a — Filter kecocokan tepat untuk jenis instance tertentu

AWS CLI

Untuk menggambarkan topologi instance dengan tipe instance tertentu

Gunakan describe-instance-topology perintah dengan `instance-type` filter. Output hanya mencakup instance dengan jenis instance yang ditentukan.

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --filters Name=instance-type,Values=trn1n.32xlarge
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menggambarkan topologi instance dengan tipe instance tertentu

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -Filter @{Name="instance-type"; Values="trn1n.32xlarge"}
```

Contoh 3b — Filter kartu liar untuk keluarga instance

AWS CLI

Untuk menggambarkan topologi instance dengan keluarga instance tertentu

Gunakan describe-instance-topology perintah dengan `instance-type` filter. Output hanya mencakup instance dengan keluarga instance yang ditentukan.

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --filters Name=instance-type,Values=trn1*
```

Berikut ini adalah output contoh.


```
{
    "Instances": [
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-3333333333example",
            "InstanceType": "trn1.32xlarge",
            "NetworkNodes": [
                "nn-1212121212example",
                "nn-1211122211example",
                "nn-1311133311example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az4",
            "AvailabilityZone": "us-west-2d"
        },
        {
            "InstanceId": "i-444444444example",
            "InstanceType": "trn1.2xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-5434334334example",
                "nn-1235301234example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
```

```
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menggambarkan topologi instance dengan keluarga instance tertentu

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -Filter @{Name="instance-type"; Values="trn1*"}
```

Contoh 3c — Keluarga instance gabungan dan filter kecocokan tepat

AWS CLI

Untuk menggambarkan topologi instance dengan keluarga instance atau tipe instance

Gunakan describe-instance-topology perintah dengan `instance-type` filter. Output hanya mencakup contoh yang memenuhi kriteria yang ditentukan.

```
aws ec2 describe-instance-topology \
    --region us-west-2 \
    --filters "Name=instance-type,Values=p4d*,trn1n.32xlarge"
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
```

```
            "AvailabilityZone": "us-west-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-4343434343example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "usw2-az2",
            "AvailabilityZone": "us-west-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menggambarkan topologi instance dengan keluarga instance atau tipe instance

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -Filter @{Name="instance-type"; Values="p4d*", "trn1n.32xlarge"}
```

## Contoh 4: DescribeInstanceTopology - Filter ID Zona

Anda dapat menggunakan filter `zone-id` untuk memfilter berdasarkan Zona Ketersediaan atau Local Zones. Anda juga dapat menggabungkan filter Zona Ketersediaan dan filter Local Zones.

Contoh 4a — Filter Zona Ketersediaan

AWS CLI

Untuk menjelaskan topologi instance di Zona Ketersediaan tertentu

Gunakan describe-instance-topology perintah dengan `zone-id` filter. Output hanya mencakup instance di Zona Ketersediaan yang ditentukan.

```
aws ec2 describe-instance-topology \
```

```
    --region us-east-1 \
    --filters Name=zone-id,Values=use1-az1
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "use1-az1",
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menjelaskan topologi instance di Zona Ketersediaan tertentu

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -Filter @{Name="zone-id"; Values="use1-az1"}
```

Contoh 4b — Filter Zona Lokal

AWS CLI

Untuk menggambarkan topologi instance di Zona Lokal tertentu

Gunakan describe-instance-topology perintah dengan `zone-id` filter. Output hanya mencakup instance di Zona Lokal yang ditentukan.

```
aws ec2 describe-instance-topology \
```

```
    --region us-east-1 \
    --filters Name=zone-id,Values=use1-atl2-az1
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "use1-atl2-az1",
            "AvailabilityZone": "us-east-1-atl-2a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menggambarkan topologi instance di Zona Lokal tertentu

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -Filter @{Name="zone-id"; Values="use1-atl2-az1"}
```

Contoh 4c — Filter Zona Ketersediaan Gabungan dan Zona Lokal

AWS CLI

Untuk menggambarkan topologi instance di zona tertentu

Gunakan describe-instance-topology perintah dengan `zone-id` filter. Output hanya mencakup instance yang berada di salah satu zona yang ditentukan.

```
aws ec2 describe-instance-topology \
    --region us-east-1 \
    --filters Name=zone-id,Values=use1-az1,use1-atl2-az1
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "use1-atl2-az1",
            "AvailabilityZone": "us-east-1-atl-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "use1-az1",
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menggambarkan topologi instance di zona tertentu

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -Filter @{Name="zone-id"; Values="use1-az1", "use1-atl2-az1"}
```

## Contoh 5: DescribeInstanceTopology - Jenis instans dan filter ID zona

Anda dapat menggabungkan filter dalam satu perintah.

AWS CLI

Untuk menjelaskan topologi instance dengan tipe instans tertentu, keluarga instans, dan zona

Gunakan describe-instance-topology perintah dengan `zone-id` filter `instance-type` dan. Respons berisi instance apa pun dengan salah satu jenis instance yang ditentukan dan berada di salah satu zona yang ditentukan.

```
aws ec2 describe-instance-topology \
    --region us-east-1 \
    --filters "Name=instance-type,Values=p4d*,trn1n.32xlarge" \
              "Name=zone-id,Values=use1-az1,use1-atl2-az1"
```

Berikut ini adalah output contoh.

```
{
    "Instances": [
        {
            "InstanceId": "i-1111111111example",
            "InstanceType": "p4d.24xlarge",
            "GroupName": "ML-group",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3333333333example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "use1-atl2-az1",
            "AvailabilityZone": "us-east-1-atl-2a"
        },
        {
            "InstanceId": "i-2222222222example",
            "InstanceType": "trn1n.32xlarge",
            "NetworkNodes": [
```

```
                "nn-1111111111example",
                "nn-2222222222example",
                "nn-3214313214example"
            ],
            "CapacityBlockId": "null",
            "ZoneId": "use1-az1",
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menjelaskan topologi instance dengan tipe instans tertentu, keluarga instans, dan zona

Gunakan Get-EC2InstanceTopology cmdlet.

```
Get-EC2InstanceTopology `
    -Filter @{Name="instance-type"; Values="p4d*", "trn1n.32xlarge"} `
            @{Name="zone-id"; Values="use1-az1", "use1-atl2-az1"}
```

## Contoh 6: DescribeCapacityReservationTopology - ID Reservasi Kapasitas

AWS CLI

Untuk menjelaskan topologi Reservasi Kapasitas tertentu

Gunakan describe-capacity-reservation-topology perintah dengan `capacity-reservation-id` parameter. Output hanya mencakup Reservasi Kapasitas yang cocok dengan ID Reservasi Kapasitas yang ditentukan.

```
aws ec2 describe-capacity-reservation-topology \
    --region us-east-1 \
    --capacity-reservation-id cr-1111111111example cr-2222222222example
```

Berikut ini adalah output contoh.

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-1111111111example",
```

```
            "CapacityBlockId": "null",
            "State": "active",
            "InstanceType": "p5.48xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example"
            ],
            "AvailabilityZone": "us-east-1a"
        },
        {
            "CapacityReservationId": "cr-2222222222example",
            "CapacityBlockId": "null",
            "State": "active",
            "InstanceType": "p5en.48xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example"
            ],
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menjelaskan topologi Reservasi Kapasitas tertentu

Gunakan Get-EC2CapacityReservationTopology cmdlet.

```
Get-EC2CapacityReservationTopology `
    -CapacityReservationId cr-1111111111example cr-2222222222example
```

## Contoh 7: DescribeCapacityReservationTopology - Filter jenis instans

Anda dapat memfilter berdasarkan tipe instans tertentu (sama persis) atau memfilter menurut keluarga instans (menggunakan wildcard). Anda juga dapat menggabungkan filter tipe instans tertentu dan filter keluarga instans.

AWS CLI

Untuk menjelaskan topologi Reservasi Kapasitas dengan jenis instans tertentu

Gunakan describe-capacity-reservation-topology perintah dengan `instance-type` filter. Respons berisi instance apa pun dengan jenis instance yang ditentukan.

```
aws ec2 describe-capacity-reservation-topology \
    --region us-east-1 \
    --filters Name=instance-type,Values=p5en.48xlarge
```

Berikut ini adalah output contoh.

```
{
    "CapacityReservations": [
        {
            "CapacityReservationId": "cr-2222222222example",
            "CapacityBlockId": "null",
            "State": "active",
            "InstanceType": "p5en.48xlarge",
            "NetworkNodes": [
                "nn-1111111111example",
                "nn-2222222222example"
            ],
            "AvailabilityZone": "us-east-1a"
        }
    ],
    "NextToken": "SomeEncryptedToken"
}
```

PowerShell

Untuk menjelaskan topologi Reservasi Kapasitas dengan jenis instans tertentu

Gunakan Get-EC2CapacityReservationTopology cmdlet.

```
Get-EC2CapacityReservationTopology `
    -Filter @{Name="instance-type"; Values="p5en.48xlarge"}
```

# Grup penempatan untuk instans Amazon EC2 Anda

Untuk memenuhi kebutuhan beban kerja Anda, Anda dapat meluncurkan sekelompok instans EC2 yang saling bergantung ke dalam grup penempatan untuk memengaruhi penempatannya.

Tergantung tipe beban kerja, Anda dapat membuat grup penempatan menggunakan salah satu strategi penempatan berikut:

- Klaster – mengemas instans saling mendekat di dalam Zona Ketersediaan. Strategi ini memungkinkan beban kerja untuk mencapai kinerja jaringan latensi rendah yang diperlukan untuk komunikasi node-ke-node yang digabungkan erat yang khas dari aplikasi komputasi kinerja tinggi (HPC).
- Partisi — Menyebarkan instance Anda di seluruh partisi logis sehingga kelompok instance dalam satu partisi tidak berbagi perangkat keras yang mendasarinya dengan kelompok instance di partisi yang berbeda. Strategi ini biasanya digunakan oleh beban kerja yang terdistribusi dan direplikasi besar, seperti Hadoop, Cassandra, dan Kafka.
- Sebaran – secara ketat menempatkan sekelompok kecil instans di seluruh perangkat keras yang mendasari untuk mengurangi kegagalan yang berhubungan.
- Waktu presisi — Menempatkan instance pada perangkat keras yang didukung dengan akses langsung ke sumber waktu presisi tinggi dalam AWS infrastruktur.

Grup penempatan adalah opsional. Jika Anda tidak meluncurkan instans Anda ke dalam grup penempatan, EC2 mencoba menempatkan instans sedemikian rupa sehingga semua instans Anda tersebar di seluruh perangkat keras yang mendasarinya untuk meminimalkan kegagalan yang berkorelasi.

Harga

Tidak ada biaya untuk membuat grup penempatan.

Aturan dan batasan

Sebelum Anda menggunakan grup penempatan, perhatikan aturan berikut ini:

- Sebuah instance dapat ditempatkan dalam satu grup penempatan pada satu waktu; Anda tidak dapat menempatkan instance di beberapa grup penempatan.
- Anda tidak dapat menggabungkan grup penempatan.
- On-Demand Reservasi Kapasitas dan Instans Cadangan zona memungkinkan Anda untuk memesan kapasitas untuk instans EC2 di Zona Ketersediaan. Saat Anda meluncurkan instance, jika atribut instans cocok dengan atribut yang ditentukan oleh Reservasi On-Demand Kapasitas atau Instans Cadangan zona, maka kapasitas yang dicadangkan secara otomatis digunakan oleh instans. Ini juga berlaku jika Anda meluncurkan instance ke grup penempatan.

- Anda tidak dapat meluncurkan Host Khusus di grup penempatan.
- Anda tidak dapat meluncurkan Instans Spot yang dikonfigurasi untuk menghentikan atau hibernasi saat interupsi dalam grup penempatan.

Daftar Isi


- Strategi penempatan untuk grup penempatan Anda
- Membuat grup penempatan untuk instans EC2 Anda
- Mengubah penempatan untuk instance EC2
- Menghapus grup penempatan
- Grup penempatan bersama
- Grup penempatan pada AWS Outposts


## Strategi penempatan untuk grup penempatan Anda

Anda dapat membuat grup penempatan untuk instans EC2 menggunakan salah satu strategi penempatan berikut.

Strategi penempatan


- Grup penempatan klaster
- Grup penempatan partisi
- Grup penempatan tersebar
- Grup penempatan waktu presisi


### Grup penempatan klaster

Grup penempatan klaster adalah pengelompokan logis dari instans di dalam Zona Ketersediaan. Instans tidak diisolasi ke rak tunggal. Grup penempatan klaster dapat menjangkau jaringan privat virtual (VPC) yang di-peering di Wilayah yang sama. Instans dalam grup penempatan cluster yang sama menikmati batas throughput per aliran yang lebih tinggi untuk TCP/IP lalu lintas dan ditempatkan di segmen bandwidth biseksi tinggi yang sama dari jaringan.

Image berikut menunjukkan instans yang ditempatkan dalam grup penempatan klaster.




Grup penempatan klaster direkomendasikan untuk aplikasi yang mendapatkan keuntungan dari latensi jaringan rendah, throughput jaringan tinggi, atau keduanya. Grup-grup itu juga direkomendasikan ketika mayoritas lalu lintas jaringan berada di antara instans dalam grup. Untuk menyediakan latensi terendah dan performa jaringan paket-per-detik tertinggi untuk grup penempatan Anda, pilih tipe instans yang mendukung peningkatan jaringan. Untuk informasi lebih lanjut, lihat Jaringan yang Ditingkatkan.

Anda dapat memastikan bahwa instans yang diluncurkan dalam grup penempatan cluster ditempatkan pada perangkat keras dengan kemampuan waktu presisi dengan menentukan grup penempatan waktu presisi induk saat Anda membuat grup penempatan cluster. Gunakan `--parent-group-id` parameter saat Anda membuat grup penempatan cluster untuk membangun hubungan ini.

Kami menyarankan Anda untuk meluncurkan instans Anda dengan cara berikut:

- Gunakan permintaan peluncuran tunggal untuk meluncurkan jumlah instans yang Anda butuhkan dalam grup penempatan.
- Gunakan tipe instans yang sama untuk semua instans di grup penempatan.

Jika Anda mencoba menambahkan lebih banyak instans ke grup penempatan nanti, atau jika Anda mencoba meluncurkan lebih dari satu tipe instans dalam grup penempatan, Anda meningkatkan peluang mendapatkan kesalahan kapasitas yang tidak cukup.

Jika Anda menghentikan satu instans dalam grup penempatan dan kemudian memulainya lagi, ini masih berjalan dalam grup penempatan. Namun demikian, momen mulai gagal jika tidak cukup kapasitas untuk instans.

Jika Anda menerima kesalahan kapasitas saat meluncurkan suatu instans dalam grup penempatan yang sudah memiliki instans, hentikan dan mulai semua instans dalam grup penempatan, dan coba luncurkan lagi. Memulai instance dapat memigrasinya ke perangkat keras yang memiliki kapasitas untuk semua instans yang diminta.

Aturan dan batasan

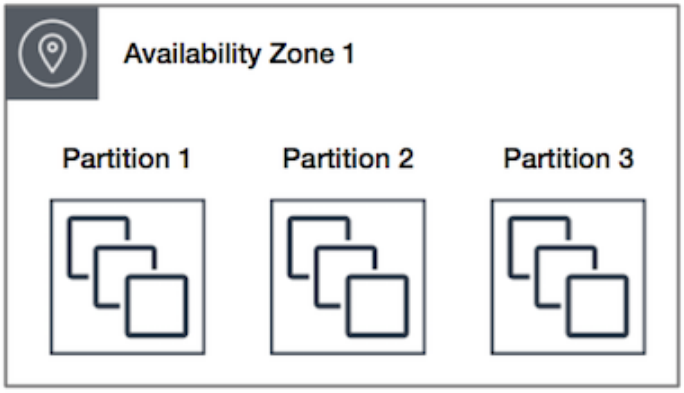
Aturan berikut berlaku untuk grup penempatan klaster:

- Berikut ini adalah tipe instans yang didukung:
  - Instans generasi saat ini, kecuali untuk inst ans kinerja yang dapat meledak (misalnya, T2), instans Mac1, dan M7i-flex instans.
  - Instans generasi sebelumnya berikut: A1, C3, C4, I2, M4, R3, dan R4.
- Grup penempatan klaster tidak dapat mencakup beberapa Zona Ketersediaan.
- Kecepatan throughput maksimum jaringan di antara dua instans dalam grup penempatan klaster dibatasi oleh pelambatan dua instans. Untuk aplikasi dengan persyaratan high-throughput, pilih tipe instans dengan sambungan jaringan yang memenuhi kebutuhan Anda.
- Untuk instans yang diaktifkan untuk jaringan yang ditingkatkan, aturan berikut berlaku:
  - Instans di dalam grup penempatan klaster dapat menggunakan hingga 10 Gbps untuk lalu lintas alur tunggal. Instans yang tidak berada di dalam grup penempatan klaster dapat menggunakan hingga 5 Gbps untuk lalu lintas alur tunggal.
  - Lalu lintas ke dan dari bucket Amazon S3 dalam Wilayah yang sama melalui ruang alamat IP publik atau melalui titik akhir VPC dapat menggunakan semua bandwidth agregat instans yang tersedia.
- Anda dapat meluncurkan beberapa tipe instans ke dalam grup penempatan klaster. Namun demikian, ini mengurangi kemungkinan bahwa jadwal yang diperlukan akan tersedia untuk peluncuran Anda agar berhasil. Kami menyarankan Anda menggunakan tipe instans yang sama untuk semua instans dalam grup penempatan klaster.
- Sebaiknya Anda mencadangkan kapasitas secara eksplisit dalam grup penempatan cluster dengan membuat Reservasi On-Demand Kapasitas di grup penempatan cluster. Perhatikan bahwa Anda tidak dapat memesan kapasitas menggunakan Instans Cadangan zona, karena mereka tidak dapat mencadangkan kapasitas secara eksplisit dalam grup penempatan.
- Blok Kapasitas tidak mendukung grup penempatan. Untuk informasi selengkapnya, lihat Blok Kapasitas untuk ML.
- Lalu lintas jaringan ke internet dan melalui Direct Connect koneksi ke sumber daya lokal dibatasi hingga 5 Gbps untuk grup penempatan cluster.

## Grup penempatan partisi

Grup penempatan partisi membantu mengurangi kemungkinan kegagalan perangkat keras terkait untuk aplikasi Anda. Saat menggunakan partisi grup penempatan, Amazon EC2 membagi setiap grup ke dalam segmen logis yang disebut partisi. Amazon EC2 memastikan bahwa setiap partisi di dalam grup penempatan memiliki set rak sendiri. Setiap rak IT jaringan dan sumber daya sendiri. Tidak ada dua bagian di dalam grup penempatan yang memiliki rak yang sama, yang memungkinkan Anda mengisolasi dampak kegagalan perangkat keras di dalam aplikasi Anda.

Image berikut adalah representasi visual sederhana dari grup penempatan partisi dalam satu Zona Ketersediaan. Ini menunjukkan instans yang ditempatkan ke dalam grup penempatan partisi dengan tiga partisi—Partisi 1, Partisi 2, dan Partisi 3. Setiap partisi terdiri dari beberapa instans. Instans dalam suatu partisi tidak menggunakan bersama rak dengan instans di dalam partisi lainnya, yang memungkinkan Anda untuk mencakup dampak kegagalan perangkat lunak tunggal ke hanya partisi terkait.




Grup penempatan partisi dapat digunakan untuk menerapkan beban kerja terdistribusi dan tereplikasi besar, seperti HDFS, HBase, dan Cassandra, di seluruh rak yang berbeda. Saat Anda meluncurkan instans ke dalam grup penempatan partisi, Amazon EC2 mencoba mendistribusikan instans secara merata ke jumlah partisi yang Anda tentukan. Anda juga dapat meluncurkan instans ke partisi tertentu untuk memiliki lebih banyak kontrol terhadap lokasi instans.

Suatu grup penempatan partisi dapat memiliki partisi pada beberapa Zona Ketersediaan di Wilayah yang sama. Grup penempatan partisi dapat memiliki maksimum tujuh partisi per Zona Ketersediaan. Jumlah instans yang dapat diluncurkan ke grup penempatan partisi hanya dibatasi oleh batas akun Anda.

Selain itu, grup penempatan partisi menawarkan visibilitas ke partisi—Anda dapat melihat instance mana yang berada di partisi mana. Anda dapat membagikan informasi ini dengan aplikasi topology-

aware, seperti HDFS, HBase, dan Cassandra. Aplikasi ini menggunakan informasi ini untuk membuat keputusan replikasi data cerdas guna meningkatkan ketersediaan data durabilitas data.

Jika Anda memulai atau meluncurkan suatu instans dalam grup penempatan partisi dan tidak ada perangkat keras unik yang cukup untuk memenuhi permintaan, maka permintaan tersebut gagal. Amazon EC2 menyediakan perangkat keras yang lebih berbeda dari waktu ke waktu, sehingga Anda dapat mencoba permintaan Anda lagi nanti.

Aturan dan batasan

Aturan berikut berlaku untuk grup penempatan partisi:

- Grup penempatan partisi mendukung memiliki maksimum tujuh partisi per Zona Ketersediaan. Jumlah instans yang dapat diluncurkan dalam grup penempatan partisi hanya dibatasi oleh batas akun Anda.
- Saat Anda meluncurkan instans ke dalam grup penempatan partisi, Amazon EC2 mencoba mendistribusikan instans tersebut secara merata ke semua partisi. Amazon EC2 tidak menjamin distribusi instans secara merata ke semua partisi.
- Sebuah grup penempatan partisi dengan Instans Khusus bisa memiliki maksimum dua partisi.
- Reservasi Kapasitas tidak menyimpan kapasitas dalam grup penempatan partisi.

## Grup penempatan tersebar

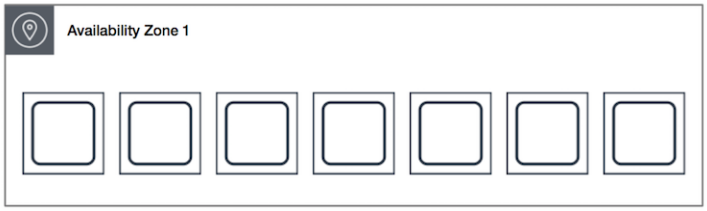
Grup penempatan sebaran adalah kelompok instans yang ditempatkan pada perangkat keras yang berbeda.

Grup penempatan sebaran direkomendasikan untuk aplikasi yang memiliki sejumlah kecil instans penting yang harus disimpan terpisah satu sama lain. Meluncurkan instance dalam grup penempatan spread mengurangi risiko kegagalan simultan yang mungkin terjadi ketika instans berbagi peralatan yang sama. Grup penempatan spread menyediakan akses ke perangkat keras yang berbeda, dan karenanya cocok untuk mencampur jenis instans atau meluncurkan instans dari waktu ke waktu.

Jika Anda memulai atau meluncurkan suatu instans dalam grup penempatan sebaran dan tidak ada perangkat keras unik yang cukup untuk memenuhi permintaan, maka permintaan tersebut gagal. Amazon EC2 menyediakan perangkat keras yang lebih berbeda dari waktu ke waktu, sehingga Anda dapat mencoba permintaan Anda lagi nanti.

Grup penempatan spread dapat menyebarkan instance di seluruh rak atau host. Untuk informasi selengkapnya, lihat the section called “Grup penempatan pada AWS Outposts”.

Gambar berikut menunjukkan tujuh contoh dalam grup penempatan spread dalam satu Zona Ketersediaan. Setiap instance ditempatkan pada rak yang berbeda, di mana setiap rak memiliki jaringan dan sumber daya sendiri.




Aturan dan batasan

Aturan berikut berlaku untuk grup penempatan tersebar:

- Rack-level grup penempatan spread didukung di Wil AWS ayah dan seterusnya AWS Outposts.
- Host-level grup penempatan spread hanya didukung pada AWS Outposts.
- Grup penempatan spread tingkat rak dapat menjangkau beberapa Zona Ketersediaan di Wilayah yang sama, tetapi mendukung maksimal tujuh instans yang sedang berjalan di setiap Zona Ketersediaan.

  Misalnya, di Wilayah dengan tiga Zona Ketersediaan, Anda dapat menjalankan total 21 instans dalam grup penempatan spread, dengan tujuh instans di setiap Zona Ketersediaan. Jika Anda mencoba memulai instans kedelapan di Zona Ketersediaan yang sudah memiliki tujuh instans, instans tidak diluncurkan.

  Jika Anda membutuhkan lebih dari tujuh instans di Zona Ketersediaan, sebaiknya Anda menggunakan beberapa grup penempatan sebaran. Menggunakan beberapa grup penempatan spread tidak menjamin bahwa instance tersebar di seluruh grup, tetapi memastikan mereka tersebar dalam setiap grup, yang membatasi dampak dari kelas kegagalan tertentu.
- Pada AWS Outposts, grup penempatan spread tingkat rak dapat menampung instans sebanyak yang Anda miliki di penerapan Outpost, dan grup penempatan spread tingkat host dapat menampung instans sebanyak yang Anda miliki host.
- Grup penempatan sebaran tidak didukung untuk Instans Khusus.
- Reservasi Kapasitas tidak menyimpan kapasitas dalam grup penempatan sebaran.

## Grup penempatan waktu presisi

Grup penempatan waktu presisi menempatkan instance pada perangkat keras yang didukung dengan akses langsung ke sumber waktu presisi tinggi dalam AWS infrastruktur. Instans yang diluncurkan dalam grup penempatan waktu presisi dapat mengakses Layanan Sinkronisasi Waktu Amazon yang disempurnakan, yang menyediakan sumber waktu lokal dengan akurasi lebih tinggi dibandingkan dengan titik akhir NTP lokal standar yang tersedia untuk semua instans Amazon EC2.

Grup penempatan waktu presisi direkomendasikan untuk aplikasi yang memerlukan sinkronisasi jam akurat mikrodetik, seperti database terdistribusi yang memerlukan pemesanan dan konsistensi transaksi, layanan keuangan yang memerlukan stempel waktu yang tepat, dan sistem terdistribusi yang memesan peristiwa menggunakan pembacaan jam lokal.

Contoh dalam grup penempatan waktu presisi mendapat manfaat dari hal berikut:

- Sumber waktu NTP lokal yang ditingkatkan dengan akurasi yang ditingkatkan, tersedia untuk semua sistem operasi yang didukung tanpa konfigurasi tambahan.
- Akses ke perangkat Jam Perangkat Keras PTP (PHC) untuk sinkronisasi akurat mikrodetik (hanya instans Linux).
- Timestamp paket perangkat keras untuk pengukuran jaringan resolusi nanodetik (hanya instans Linux).

Untuk informasi tentang mengonfigurasi instance Anda untuk menggunakan sumber waktu yang disempurnakan, lihatMengatur referensi waktu pada instans EC2 Anda untuk menggunakan Layanan Sinkronisasi Waktu Amazon lokal.

Aturan berikut berlaku untuk grup penempatan waktu presisi:

Aturan dan batasan

- Untuk daftar keluarga instans yang didukung saat ini, lihatMengatur referensi waktu pada instans EC2 Anda untuk menggunakan Layanan Sinkronisasi Waktu Amazon lokal.
- Jika Anda memulai atau meluncurkan instance dalam grup penempatan waktu presisi dan perangkat keras tidak mencukupi untuk menyediakan akses ke Layanan Sinkronisasi Waktu Amazon yang disempurnakan, permintaan gagal. Coba lagi nanti atau coba Zona Ketersediaan yang berbeda.
- Anda tidak dapat menghapus grup penempatan waktu presisi yang merupakan induk dari grup penempatan cluster.

- Tidak ada biaya tambahan untuk menggunakan grup penempatan waktu presisi.

## Membuat grup penempatan untuk instans EC2 Anda

Anda dapat menggunakan grup penempatan untuk mengontrol penempatan instance relatif satu sama lain. Setelah membuat grup penempatan, Anda dapat meluncurkan instance di grup penempatan.

Batasan

Anda dapat membuat maksimum 500 grup penempatan per Wilayah.

Console

Untuk membuat grup penempatan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Grup Penempatan.
3. Pilih Buat grup penempatan.
4. Untuk Nama, tentukan nama untuk grup.
5. Untuk Strategi Pen empatan, pilih strategi penempatan untuk grup: Cluster, S pread, Partition, atau Precision time.
   a. Jika Anda memilih S pread, untuk Lev el S pread, pilih Host atau Rack.
   b. Jika Anda memilih Partisi, untuk Jumlah partisi, masukkan jumlah partisi untuk grup.
   c. Jika Anda memilih Cluster, Anda dapat secara opsional menentukan grup penempatan waktu presisi induk untuk memastikan bahwa instance dalam grup penempatan cluster diluncurkan pada perangkat keras yang mampu waktu presisi.
6. (Opsional) Untuk menambahkan tag, di bawah Tag, pilih Tambahkan tag baru, lalu masukkan kunci dan nilai.
7. Pilih Buat grup.

AWS CLI

Gunakan perintah create-placement-group.

Untuk membuat grup penempatan cluster

Contoh berikut membuat grup pen `cluster` empatan yang menggunakan strategi penempatan, dan menerapkan tag dengan kunci `purpose` dan nilai`production`.

```
aws ec2 create-placement-group \
    --group-name my-cluster \
    --strategy cluster \
    --tag-specifications 'ResourceType=placement-
group,Tags={Key=purpose,Value=production}'
```

Untuk membuat grup penempatan cluster dengan induk grup penempatan waktu presisi

Contoh berikut membuat grup penempatan cluster yang menentukan grup penempatan waktu presisi induk menggunakan `--parent-group-id` parameter.

```
aws ec2 create-placement-group \
    --group-name my-cluster \
    --strategy cluster \
    --parent-group-id pg-0aaa1111111111111
```

Untuk membuat grup penempatan partisi

Contoh berikut membuat grup penempatan yang menggunakan strategi pen `partition` empatan, dan menentukan lima partisi menggunakan `--partition-count` parameter.

```
aws ec2 create-placement-group \
    --group-name HDFS-Group-A \
    --strategy partition \
    --partition-count 5
```

Untuk membuat grup penempatan waktu presisi

Contoh berikut membuat grup penempatan yang menggunakan strategi pen `precision-time` empatan.

```
aws ec2 create-placement-group \
    --group-name my-precision-time-pg \
    --strategy precision-time
```

PowerShell

Gunakan New-EC2PlacementGroup cmdlet.

Untuk membuat grup penempatan cluster

Contoh berikut membuat grup penempatan cluster.

```
New-EC2PlacementGroup `
    -GroupName my-placement-group `
    -Strategy cluster
```

Untuk membuat grup penempatan cluster dengan induk grup penempatan waktu presisi

Contoh berikut membuat grup penempatan cluster yang menentukan grup penempatan waktu presisi induk menggunakan `-ParentGroupId` parameter.

```
New-EC2PlacementGroup `
    -GroupName my-placement-group `
    -Strategy cluster `
    -ParentGroupId pg-0aaa1111111111111
```

Untuk membuat grup penempatan partisi

Contoh berikut membuat grup penempatan partisi.

```
New-EC2PlacementGroup `
    -GroupName my-placement-group `
    -Strategy partition `
    -PartitionCount 5
```

Untuk membuat grup penempatan waktu presisi

Contoh berikut membuat grup penempatan waktu presisi.

```
New-EC2PlacementGroup `
    -GroupName my-precision-time-pg `
    -Strategy precision-time
```

## Mengubah penempatan untuk instance EC2

Anda dapat mengubah grup penempatan suatu instans dengan cara berikut:

- Menambahkan instance ke grup penempatan
- Pindahkan satu instans dari satu grup penempatan ke grup penempatan lainnya

- Menghapus instans dari grup penempatan

### Persyaratan

Sebelum Anda dapat mengubah grup penempatan untuk sebuah instance, instance harus berada dalam `stopped` status.

### Console

Untuk mengubah penempatan instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih Tindakan, Pengaturan instans, Modifikasi penempatan instans.
5. Untuk grup Pen empatan, lakukan salah satu hal berikut:
   - Untuk menambahkan instance ke grup penempatan, pilih grup penempatan.
   - Untuk memindahkan instance dari satu grup penempatan ke grup penempatan lainnya, pilih grup penempatan.
   - Untuk menghapus instance dari grup penempatan, pilih Tidak Ada.
6. Pilih Simpan.

### AWS CLI

Untuk memindahkan instans ke grup penempatan

Gunakan perintah modify- instance-placement berikut.

```
aws ec2 modify-instance-placement \
    --instance-id i-0123a456700123456 \
    --group-name MySpreadGroup
```

Untuk menghapus instans dari grup penempatan

Gunakan perintah modify- instance-placement berikut. Ketika Anda menentukan string kosong untuk nama grup penempatan, ini menghapus instance dari grup penempatan saat ini.

```
aws ec2 modify-instance-placement \
```

```
    --instance-id i-0123a456700123456 \
    --group-name ""
```

PowerShell

Untuk memindahkan instans ke grup penempatan

Gunakan Edit-EC2InstancePlacement cmdlet dengan nama grup penempatan.

```
Edit-EC2InstancePlacement `
    -InstanceId i-0123a456700123456 `
    -GroupName MySpreadGroup
```

Untuk menghapus instans dari grup penempatan

Gunakan Edit-EC2InstancePlacement cmdlet dengan string kosong untuk nama grup penempatan.

```
Edit-EC2InstancePlacement `
    -InstanceId i-0123a456700123456 `
    -GroupName ""
```

## Menghapus grup penempatan

Jika Anda perlu mengganti grup penempatan atau tidak lagi memerlukannya, Anda dapat menghapusnya. Sebelum Anda dapat menghapus grup penempatan, grup penempatan harus tidak berisi instans. Anda dapat menghentikan instance, memindahkannya ke grup penempatan lain, atau menghapusnya dari grup penempatan.

Anda tidak dapat menghapus grup penempatan yang merupakan induk dari grup penempatan cluster. Hapus grup penempatan cluster terlebih dahulu.

Console

Untuk menghapus grup penempatan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Grup Penempatan.
3. Pilih grup penempatan dan pilih Tindakan, Hapus.
4. Saat diminta konfirmasi, masukkan **Delete**, lalu pilih Hapus.

AWS CLI

Untuk menghapus grup penempatan

Gunakan perintah delete-plac ement-group.

```
aws ec2 delete-placement-group --group-name my-cluster
```

PowerShell

Untuk menghapus grup penempatan

Gunakan Remove-EC2PlacementGroup cmdlet.

```
Remove-EC2PlacementGroup -GroupName my-cluster
```

## Grup penempatan bersama

Berbagi grup penempatan memungkinkan Anda memengaruhi penempatan instans yang saling bergantung yang dimiliki oleh terpisah Akun AWS. Pemilik dapat berbagi grup penempatan di beberapa Akun AWS atau dalam organisasi mereka. Peserta dapat meluncurkan instance dalam grup penempatan yang dibagikan dengan akun mereka.

Pemilik grup penempatan dapat berbagi grup penempatan dengan:

- Spesi Akun AWS fik di dalam atau di luar organisasinya
- Unit organisasi di dalam organisasi -nya
- Seluruh organisasi -nya

Anda dapat menggunakan peering VPC untuk menghubungkan instans yang dimiliki oleh terpisah Akun AWS dan mendapatkan manfaat latensi penuh yang ditawarkan oleh grup penempatan cluster bersama.

Daftar Isi


- Aturan dan batasan
- Izin yang diperlukan
- Berbagi di seluruh Availability Zone
- Berbagi kelompok penempatan

- Grup penempatan berhenti berbagi

## Aturan dan batasan

Aturan dan batasan berikut berlaku saat Anda berbagi grup penempatan atau ketika grup penempatan dibagikan dengan Anda.

- Untuk berbagi grup penempatan, Anda harus memilikinya di grup Anda Akun AWS. Anda tidak dapat membagikan grup penempatan yang telah dibagikan dengan Anda.
- Ketika Anda berbagi partisi atau grup penempatan sebaran, batas grup penempatan tidak berubah. Grup penempatan partisi bersama mendukung maksimal tujuh partisi per Zona Ketersediaan, dan grup penempatan sebaran bersama mendukung maksimal tujuh instans yang berjalan per Zona Ketersediaan.
- Untuk berbagi grup penempatan dengan organisasi atau unit organisasi di organisasi Anda, Anda harus mengaktifkan berbagi dengan AWS Organizations. Untuk informasi lebih lanjut, lihat Berbagi sumber daya AWS Anda.
- Saat menggunakan Konsol Manajemen AWS untuk meluncurkan instance, Anda dapat memilih grup penempatan apa pun yang dibagikan dengan Anda. Saat menggunakan AWS CLI untuk meluncurkan instance, Anda harus menentukan grup penempatan bersama berdasarkan ID, bukan berdasarkan nama. Anda dapat menggunakan nama grup penempatan hanya jika Anda adalah pemilik grup penempatan bersama.
- Anda bertanggung jawab untuk mengelola instans yang dimiliki oleh Anda dalam grup penempatan bersama.
- Anda tidak dapat melihat atau mengubah instans dan reservasi kapasitas yang terkait dengan grup penempatan bersama tetapi tidak dimiliki oleh Anda.
- Nama Sumber Daya Amazon (ARN) dari grup penempatan berisi ID akun yang memiliki grup penempatan. Anda dapat menggunakan bagian ID akun dari grup penempatan ARN untuk mengidentifikasi pemilik grup penempatan yang dibagikan dengan Anda.

## Izin yang diperlukan

Untuk berbagi grup penempatan, pengguna harus memiliki izin untuk tindakan berikut:

- `ec2:PutResourcePolicy`
- `ec2:DeleteResourcePolicy`

## Berbagi di seluruh Availability Zone

Untuk memastikan bahwa sumber daya didistribusikan di seluruh Zona Ketersediaan untuk suatu Wilayah, kami secara independen memetakan Zona Ketersediaan ke nama untuk setiap akun. Hal ini dapat menyebabkan perbedaan penamaan Zona Ketersediaan di seluruh akun. Misalnya, Zona Ketersediaan `us-east-1a` untuk Anda Akun AWS mungkin tidak memiliki lokasi yang sama dengan `us-east-1a` yang lain Akun AWS.

Untuk menentukan lokasi Host Khusus relatif terhadap akun Anda, Anda harus menggunakan ID Zona Ketersediaan (ID AZ). ID AZ adalah pengidentifikasi unik dan konsisten untuk Zona Ketersediaan di semua akun AWS . Misalnya, `use1-az1` adalah ID Zona Ketersediaan untuk Wilayah `us-east-1` dan lokasinya sama di setiap akun AWS . Untuk informasi selengkapnya, lihat ID AZ.

## Berbagi kelompok penempatan

Untuk membagikan grup penempatan, Anda harus menambahkannya ke berbagi sumber daya. Berbagi sumber daya adalah AWS RAM sumber daya yang memungkinkan Anda berbagi sumber daya Akun AWS. Pembagian sumber daya menentukan sumber daya yang akan dibagikan, dan konsumen yang akan berbagi dengan mereka.

Jika Anda adalah bagian dari organisasi di AWS Organizations, berbagi dalam organisasi Anda diaktifkan, dan konsumen di organisasi Anda diberikan akses ke grup penempatan bersama.

Jika grup penempatan dibagikan dengan pihak Akun AWS luar organisasi Anda, Akun AWS pemilik menerima undangan untuk bergabung dengan pembagian sumber daya. Mereka dapat mengakses grup penempatan bersama setelah menerima undangan.

Anda dapat berbagi grup penempatan di seluruh Akun AWS penggunaan AWS Resource Access Manager. Untuk informasi selengkapnya, lihat Membuat berbagi sumber daya di Panduan AWS RAM Pengguna.

## Grup penempatan berhenti berbagi

Pemilik grup penempatan dapat membatalkan pembagian grup penempatan bersama kapan saja. Saat Anda membatalkan berbagi grup penempatan bersama, perubahan berikut terjadi:

- Grup penempatan yang digunakan bersama tidak lagi dapat meluncurkan instans atau cadangan kapasitas. Akun AWS

- Instans apa pun yang berjalan di grup penempatan bersama dipisahkan dari grup penempatan, tetapi tetap berjalan di grup penempatan Anda Akun AWS.
- Setiap reservasi kapasitas dalam grup penempatan bersama dipisahkan dari grup penempatan, tetapi tetap tersedia untuk Anda Akun AWS di grup penempatan.

Untuk informasi selengkapnya, lihat Menghapus berbagi sumber daya di Panduan AWS RAM Pengguna.

## Grup penempatan pada AWS Outposts

AWS Outposts adalah layanan yang dikelola sepenuhnya yang memperluas AWS infrastruktur, layanan, API, dan alat ke tempat pelanggan. Dengan menyediakan akses lokal ke infrastruktur AWS terkelola, AWS Outposts memungkinkan pelanggan untuk membangun dan menjalankan aplikasi di tempat menggunakan antarmuka pemrograman yang sama seperti di Wil AWS ayah, sambil menggunakan sumber daya komputasi dan penyimpanan lokal untuk latensi yang lebih rendah dan kebutuhan pemrosesan data lokal.

Outpost adalah kumpulan kapasitas AWS komputasi dan penyimpanan yang digunakan di situs pelanggan. AWS mengoperasikan, memantau, dan mengelola kapasitas ini sebagai bagian dari Wil AWS ayah.

Anda dapat membuat grup penempatan di Outposts yang telah Anda buat di akun Anda. Hal ini memungkinkan Anda untuk menyebarkan instans di perangkat keras yang mendasarinya di Outpost di situs Anda. Anda membuat dan menggunakan grup penempatan di Outposts dengan cara yang sama seperti Anda membuat dan menggunakan grup penempatan di Zona Ketersediaan biasa. Saat Anda membuat grup penempatan dengan strategi penyebaran di Outpost, Anda dapat memilih agar grup penempatan menyebarkan instans di seluruh host atau rak. Menyebarkan instance di seluruh host memungkinkan Anda menggunakan strategi penyebaran dengan Outpost rak tunggal.

Pertimbangan-pertimbangan

- Grup penempatan spread tingkat rak dapat menampung instans sebanyak yang Anda miliki di penerapan Outpost Anda.
- Grup penempatan spread tingkat host dapat menampung instans sebanyak yang Anda miliki host dalam penerapan Outpost Anda.

Prasyarat

Anda harus menginstal Outpost di situs Anda. Untuk informasi selengkapnya, lihat Membuat Outpost dan memesan kapasitas Outpost di Panduan Pengguna AWS Outposts .

Untuk menggunakan grup penempatan di Outpost

1. Buatlah subnet pada Outpost. Untuk informasi selengkapnya, lihat Membuat subnet di Panduan Pengguna AWS Outposts .
2. Buat grup penempatan di Wilayah terkait Outpost. Jika Anda membuat grup penempatan dengan strategi spread, Anda dapat memilih spread tingkat host atau rak untuk menentukan bagaimana grup menyebarkan instance di seluruh perangkat keras yang mendasarinya di Outpost Anda. Untuk informasi selengkapnya, lihat the section called “Buat grup penempatan”.
3. Luncurkan instans ke dalam grup penempatan. Untuk Subnet pilih subnet yang Anda buat di Langkah 1, dan untuk Nama grup penempatan, pilih grup penempatan yang Anda buat di Langkah 2. Untuk informasi selengkapnya, lihat Meluncurkan instans pada Outpost di Panduan Pengguna AWS Outposts .

# Maximum transmission unit (MTU) jaringan untuk instans EC2 Anda

Maximum transmission unit (MTU) dari koneksi jaringan adalah ukuran, dalam byte, dari paket terbesar yang dapat diizinkan yang dapat dilewatkan melalui koneksi. Semakin besar MTU suatu koneksi, semakin banyak data yang dapat dilewatkan dalam satu paket tunggal. Paket Ethernet terdiri dari frame, atau data aktual yang Anda kirim, dan informasi overhead jaringan di sekitarnya.

Frame Ethernet bisa hadir dalam format yang berbeda, dan format yang paling umum adalah format frame Ethernet v2 standar. Ini mendukung 1500 MTU, yang merupakan ukuran paket Ethernet terbesar yang didukung di hampir seluruh internet. MTU dengan dukungan maksimum untuk suatu instans bergantung pada tipe instans-nya.

Semua jenis instans EC2 mendukung 1500 MTU.

Daftar Isi


- Frame jumbo (9001 MTU)
- Path MTU Discovery
- Mengatur MTU untuk instans Amazon EC2 Anda
- Pemecahan Masalah

# Frame jumbo (9001 MTU)

Dengan jumbo frame, Anda dapat meningkatkan ukuran payload per paket, sehingga meningkatkan persentase paket yang bukan overhead paket. Dengan frame jumbo, lebih sedikit paket yang diperlukan untuk mengirim jumlah data yang dapat digunakan yang sama. Namun, jenis lalu lintas tertentu tunduk pada muatan maksimum berikut:

Batas MTU 1500 byte

- Lalu lintas melalui gateway internet
- Lalu lintas melalui koneksi VPN
- Lalu lintas AWS antar Wilayah, kecuali gateway transit digunakan

Batas MTU 8500 byte

- Lalu lintas melalui koneksi peering VPC antar wilayah

Jika paket melebihi batas MTU mereka, mereka terfragmentasi, atau mereka dijatuhkan jika `Don't Fragment` bendera diatur di header IP.

Frame Jumbo harus digunakan dengan hati-hati untuk lalu lintas internet-bound atau setiap lalu lintas yang meninggalkan VPC. Paket difragmentasi oleh sistem menengah, yang memperlambat lalu lintas ini. Untuk menggunakan frame jumbo di dalam VPC dan tidak memperlambat lalu lintas yang terikat di luar VPC, Anda dapat mengonfigurasi ukuran MTU berdasarkan rute, atau menggunakan beberapa antarmuka elastic network dengan ukuran MTU yang berbeda dan rute yang berbeda.

Untuk instans-instans dengan lokasi sama dalam grup penempatan klaster, bingkai jumbo membantu mencapai throughput jaringan semaksimum mungkin, dan dianjurkan dalam kasus ini. Untuk informasi selengkapnya, lihat Grup penempatan untuk instans Amazon EC2 Anda.

Anda dapat menggunakan frame jumbo untuk lalu lintas antara VPC dan jaringan on-premise Anda melalui Direct Connect. Untuk informasi selengkapnya, dan cara memverifikasi kemampuan Jumbo Frame, lihat MTU untuk antarmuka virtual pribadi atau antarmuka virtual transit di Panduan Direct Connect Pengguna.

Semua jenis instance generasi saat ini mendukung bingkai jumbo. Jenis instance generasi sebelumnya berikut mendukung frame jumbo: A1, C3, I2, M3, dan R3.

Sumber daya terkait

- Untuk gateway NAT, lihat Dasar-dasar gateway NAT di Panduan Pengguna Amazon VPC.
- Untuk gateway transit, lihat Unit transmisi maksimum di Panduan Pengguna Gerbang Transit Amazon VPC.
- Untuk Local Zones, lihat Pertimbangan di Panduan Pengguna AWS Local Zones.
- Untuk AWS Wavelength, lihat Unit transmisi maksimum di Panduan AWS Wavelength Pengguna.
- Untuk Pos Terdepan, lihat Persyaratan unit transmisi maksimum tautan layanan di Panduan AWS Outposts Pengguna.

## Path MTU Discovery

Penemuan MTU Jalur (PMTUD) digunakan untuk menentukan jalur MTU antara dua perangkat. Jalur MTU adalah ukuran paket maksimum yang didukung pada jalur antara host asal dan host penerima. Ketika ada perbedaan dalam ukuran MTU dalam jaringan antara dua host, PMTUD memungkinkan host penerima untuk menanggapi host asal dengan pesan ICMP. Pesan ICMP ini menginstruksikan host asal untuk menggunakan ukuran MTU terendah di sepanjang jalur jaringan dan untuk mengirim ulang permintaan. Tanpa negosiasi ini, paket drop dapat terjadi karena permintaan terlalu besar untuk diterima oleh host penerima.

Untuk IPv4, jika suatu host mengirimkan paket yang lebih besar daripada MTU host penerima atau yang lebih besar daripada MTU perangkat di sepanjang jalur, host atau perangkat penerima menjatuhkan paket, lalu mengembalikan pesan ICMP berikut: `Destination Unreachable: Fragmentation Needed and Don't Fragment was Set` (Tipe 3, Kode 4). Ini menginstruksikan host transmisi untuk membagi muatan menjadi beberapa paket yang lebih kecil, dan kemudian mentrasmisikannya kembali.

Protokol IPv6 tidak mendukung fragmentasi dalam jaringan. Jika suatu host mengirimkan paket yang lebih besar daripada MTU host penerima atau yang lebih besar daripada MTU perangkat di sepanjang jalur, host atau perangkat penerima menjatuhkan paket, lalu mengembalikan pesan ICMP berikut: `ICMPv6 Packet Too Big (PTB)` (Tipe 2). Ini menginstruksikan host transmisi untuk membagi muatan menjadi beberapa paket yang lebih kecil, dan kemudian mentrasmisikannya kembali.

Koneksi yang dilakukan melalui beberapa komponen, seperti gateway NAT dan penyeimbang beban, secara otomatis dilacak. Ini berarti bahwa pelacakan grup keamanan diaktifkan secara otomatis untuk

upaya koneksi keluar Anda. Jika koneksi dilacak secara otomatis, tidak dilacak, atau jika aturan grup keamanan Anda mengizinkan lalu lintas ICMP masuk, Anda dapat menerima tanggapan PMTUD.

Perhatikan bahwa lalu lintas ICMP dapat diblokir bahkan jika lalu lintas diizinkan di tingkat grup keamanan, seperti jika Anda memiliki entri daftar kontrol akses jaringan yang menolak lalu lintas ICMP ke subnet.

> **Important**
>
> Path MTU Discovery tidak menjamin bahwa frame jumbo tidak akan diturunkan oleh beberapa router. Gateway internet di VPC Anda akan meneruskan paket hingga 1500 byte saja. 1500 paket MTU direkomendasikan untuk lalu lintas internet.

Untuk aturan MTU atas gateway NAT, lihat Unit transmisi maksimum (MTU) di Panduan Pengguna Amazon VPC. Untuk aturan MTU atas gateway Transit, lihat Unit transmisi maksimum (MTU) di Panduan Pengguna Gateway AWS Transit.

## Mengatur MTU untuk instans Amazon EC2 Anda

Maximum transmission unit (MTU) dari koneksi jaringan adalah ukuran, dalam byte, dari paket terbesar yang dapat diizinkan yang dapat dilewatkan melalui koneksi. Semua instans Amazon EC2 mendukung frame standar (1500 MTU) dan semua jenis instans generasi saat ini mendukung jumbo frame (9001 MTU).

Anda dapat melihat MTU untuk instans Amazon EC2 Anda, melihat jalur MTU antara instans Anda dan host lain, dan mengonfigurasi instans Anda untuk menggunakan frame standar atau jumbo.

Tugas

- Periksa MTU jalur di antara dua host
- Periksa MTU untuk contoh Anda
- Mengatur MTU untuk instance Anda

### Periksa MTU jalur di antara dua host

Anda dapat memeriksa jalur MTU antara instance EC2 Anda dan host lain. Anda dapat menentukan nama DNS atau alamat IP sebagai tujuan. Jika tujuannya adalah instance EC2 lain, verifikasi bahwa grup keamanannya mengizinkan lalu lintas UDP masuk.

Prosedur yang Anda gunakan tergantung pada sistem operasi instance.

Instans Linux

Jalan tracepath kan perintah pada instance Anda untuk memeriksa jalur MTU antara instance EC2 Anda dan tujuan yang ditentukan. Perintah ini adalah bagian dari `iputils` paket, yang tersedia secara default di banyak distribusi Linux.

Contoh ini memeriksa jalur MTU antara instance EC2 dan. `amazon.com`

```
[ec2-user ~]$ tracepath amazon.com
```

Dalam contoh output ini, jalur MTU adalah 1500.

```
 1?: [LOCALHOST]      pmtu 9001
 1:  ip-172-31-16-1.us-west-1.compute.internal (172.31.16.1)   0.187ms pmtu 1500
 1:  no reply
 2:  no reply
 3:  no reply
 4:  100.64.16.241 (100.64.16.241)                              0.574ms
 5:  72.21.222.221 (72.21.222.221)                             84.447ms asymm 21
 6:  205.251.229.97 (205.251.229.97)                           79.970ms asymm 19
 7:  72.21.222.194 (72.21.222.194)                             96.546ms asymm 16
 8:  72.21.222.239 (72.21.222.239)                             79.244ms asymm 15
 9:  205.251.225.73 (205.251.225.73)                           91.867ms asymm 16
...
31:  no reply
     Too many hops: pmtu 1500
     Resume: pmtu 1500
```

Instans Windows

Untuk memeriksa jalur MTU menggunakan mturoute

1. Unduh mturoute.exe ke instans EC2 Anda dari https://elifulkerson.com/projects/mturoute.php.
2. Buka jendela Command Prompt dan ubah ke direktori untuk mengunduh mturoute.exe.
3. Gunakan perintah berikut untuk memeriksa jalur MTU antara instance EC2 Anda dan tujuan yang ditentukan. Contoh ini memeriksa jalur MTU antara instance EC2 dan. `www.elifulkerson.com`

   ```
   .\mturoute.exe www.elifulkerson.com
   ```

Dalam contoh output ini, jalur MTU adalah 1500.

```
* ICMP Fragmentation is not permitted. *
* Speed optimization is enabled. *
* Maximum payload is 10000 bytes. *
+ ICMP payload of 1472 bytes succeeded.
- ICMP payload of 1473 bytes is too big.
Path MTU: 1500 bytes.
```

## Periksa MTU untuk contoh Anda

Anda dapat memeriksa nilai MTU untuk instance Anda. Beberapa instans dikonfigurasi untuk menggunakan frame jumbo, dan lainnya dikonfigurasi untuk menggunakan ukuran frame standar.

Prosedur yang Anda gunakan tergantung pada sistem operasi instance.

Instans Linux

Untuk memeriksa pengaturan MTU di instans Linux

Jalankan ip perintah berikut pada instance EC2 Anda. Jika antarmuka jaringan utama tidak`eth0`, ganti `eth0` dengan antarmuka jaringan Anda.

```
[ec2-user ~]$ ip link show eth0
```

Dalam contoh output ini, *`mtu 9001`* menunjukkan bahwa instance menggunakan bingkai jumbo.

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc pfifo_fast state UP mode
 DEFAULT group default qlen 1000
    link/ether 02:90:c0:b7:9e:d1 brd ff:ff:ff:ff:ff:ff
```

Instans Windows

Prosedur yang Anda gunakan tergantung pada driver pada instance Anda.

ENA driver

Versi 2.1.0 dan yang lebih baru

Untuk mendapatkan nilai MTU, gunakan Get-NetAdapterAdvancedProperty perintah berikut pada instance EC2 Anda. Gunakan wildcard (asterisk) untuk mendapatkan semua nama Ethernet. Periksa output untuk nama antarmuka`*JumboPacket`. Nilai 9015 menunjukkan bahwa bingkai Jumbo diaktifkan. Bingkai jumbo dinonaktifkan secara default.

```
Get-NetAdapterAdvancedProperty -Name "Ethernet*"
```

Versi 1.5 dan sebelumnya

Untuk mendapatkan nilai MTU, gunakan Get-NetAdapterAdvancedProperty perintah berikut pada instance EC2 Anda. Periksa output untuk nama antarmukaMTU. Nilai 9001 menunjukkan bahwa bingkai Jumbo diaktifkan. Bingkai jumbo dinonaktifkan secara default.

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

Intel SRIOV 82599 driver

Untuk mendapatkan nilai MTU, gunakan Get-NetAdapterAdvancedProperty perintah berikut pada instance EC2 Anda. Periksa entri untuk nama antarmuka `*JumboPacket`. Nilai 9014 menunjukkan bahwa bingkai Jumbo diaktifkan. (Perhatikan bahwa ukuran MTU mencakup header dan muatan.) Bingkai jumbo dinonaktifkan secara default.

```
Get-NetAdapterAdvancedProperty -Name "Ethernet"
```

AWS PV driver

Untuk mendapatkan nilai MTU, gunakan perintah berikut pada instance EC2 Anda. Nama antarmuka dapat bervariasi. Dalam output, cari entri dengan nama "Ethernet," "Ethernet 2," atau "Local Area Connection". Anda akan memerlukan nama antarmuka untuk mengaktifkan atau menonaktifkan frame jumbo. Nilai 9001 menunjukkan bahwa bingkai Jumbo diaktifkan.

```
netsh interface ipv4 show subinterface
```

## Mengatur MTU untuk instance Anda

Anda mungkin ingin menggunakan bingkai jumbo untuk lalu lintas jaringan dalam VPC Anda dan bingkai standar untuk lalu lintas internet. Apa pun kasus penggunaan Anda, kami sarankan Anda memverifikasi bahwa instance Anda berperilaku seperti yang diharapkan.

Prosedur yang Anda gunakan tergantung pada sistem operasi instance.

Instans Linux

Untuk mengatur nilai MTU di instans Linux

1. Jalankan ip perintah berikut pada instance Anda. Ini menetapkan nilai MTU yang diinginkan menjadi 1500, tetapi Anda dapat menggunakan 9001 sebagai gantinya. Jika antarmuka jaringan utama tidak`eth0`, ganti `eth0` dengan antarmuka jaringan yang sebenarnya.

   ```
   [ec2-user ~]$ sudo ip link set dev eth0 mtu 1500
   ```

2. (Opsional) Untuk mempertahankan pengaturan MTU jaringan Anda setelah reboot, modifikasi file konfigurasi berikut berdasarkan tipe sistem operasi Anda.

   - Amazon Linux 2023 — U `[Link]` bah bagian file konfigurasi. File konfigurasi default adalah`/usr/lib/systemd/network/80-ec2.network`, atau Anda dapat memperbarui file konfigurasi khusus yang dibuat di run/systemd //network/, di mana nama file adalah *priority* - *interface* .network. Untuk informasi selengkapnya, lihat Layanan jaringan di dokumentasi Amazon Linux.

     ```
     MTUBytes=1500
     ```

   - Amazon Linux 2 — Tambahkan baris berikut ke `/etc/sysconfig/network-scripts/ifcfg-`*eth0* file:

     ```
     MTU=1500
     ```

     Tambahkan baris berikut ke file `/etc/dhcp/dhclient.conf`:

     ```
     request subnet-mask, broadcast-address, time-offset, routers, domain-name,
      domain-search, domain-name-servers, host-name, nis-domain, nis-servers, ntp-
     servers;
     ```

   - Distribusi Linux lainnya — Konsultasikan dokumentasi spesifik mereka.

3. (Opsional) Reboot instans Anda dan verifikasi bahwa pengaturan MTU benar.

Instans Windows

Prosedur yang Anda gunakan tergantung pada driver pada instance Anda.

ENA driver

Anda dapat mengubah MTU menggunakan Device Manager atau Set-NetAdapterAdvancedProperty perintah pada instance Anda.

Versi 2.1.0 dan yang lebih baru

Gunakan perintah berikut untuk mengaktifkan bingkai jumbo.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 9015
```

Gunakan perintah berikut untuk menonaktifkan bingkai jumbo.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 1514
```

Versi 1.5 dan sebelumnya

Gunakan perintah berikut untuk mengaktifkan bingkai jumbo.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -
RegistryValue 9001
```

Gunakan perintah berikut untuk menonaktifkan bingkai jumbo.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "MTU" -
RegistryValue 1500
```

Intel SRIOV 82599 driver

Anda dapat mengubah MTU menggunakan Device Manager atau Set-NetAdapterAdvancedProperty perintah pada instance Anda.

Gunakan perintah berikut untuk mengaktifkan bingkai jumbo.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 9014
```

Gunakan perintah berikut untuk menonaktifkan bingkai jumbo.

```
Set-NetAdapterAdvancedProperty -Name "Ethernet" -RegistryKeyword "*JumboPacket" -
RegistryValue 1514
```

AWS PV driver

Anda dapat mengubah MTU menggunakan netsh perintah pada instance Anda. Anda tidak dapat mengubah MTU menggunakan Device Manager.

Gunakan perintah berikut untuk mengaktifkan bingkai jumbo.

```
netsh interface ipv4 set subinterface "Ethernet" mtu=9001
```

Gunakan perintah berikut untuk menonaktifkan bingkai jumbo.

```
netsh interface ipv4 set subinterface "Ethernet" mtu=1500
```

## Pemecahan Masalah

Jika Anda mengalami masalah konektivitas antara instans EC2 dan cluster Amazon Redshift saat menggunakan bingkai jumbo, lihat Ku eri tampak macet dan terkadang gagal mencapai cluster di Panduan Manajemen Amazon Redshift.

# Virtual private cloud untuk instans EC2

Amazon Virtual Private Cloud (Amazon VPC) memungkinkan Anda untuk menentukan jaringan virtual di area Anda sendiri yang terisolasi secara logis di dalam AWS cloud, yang dikenal sebagai cloud pribadi virtual atau VPC. Anda dapat membuat AWS sumber daya, seperti instans Amazon EC2, ke dalam subnet VPC Anda. VPC Anda sangat menyerupai jaringan tradisional yang mungkin Anda operasikan di pusat data Anda sendiri, dengan memanfaatkan infrastruktur terukur dari AWS. Anda dapat mengonfigurasi VPC Anda; Anda dapat memilih rentang alamat IP, membuat subnet, dan mengonfigurasi tabel rute, gateway jaringan, dan pengaturan keamanan. Anda dapat menghubungkan instans dalam VPC ke internet atau ke pusat data Anda sendiri.

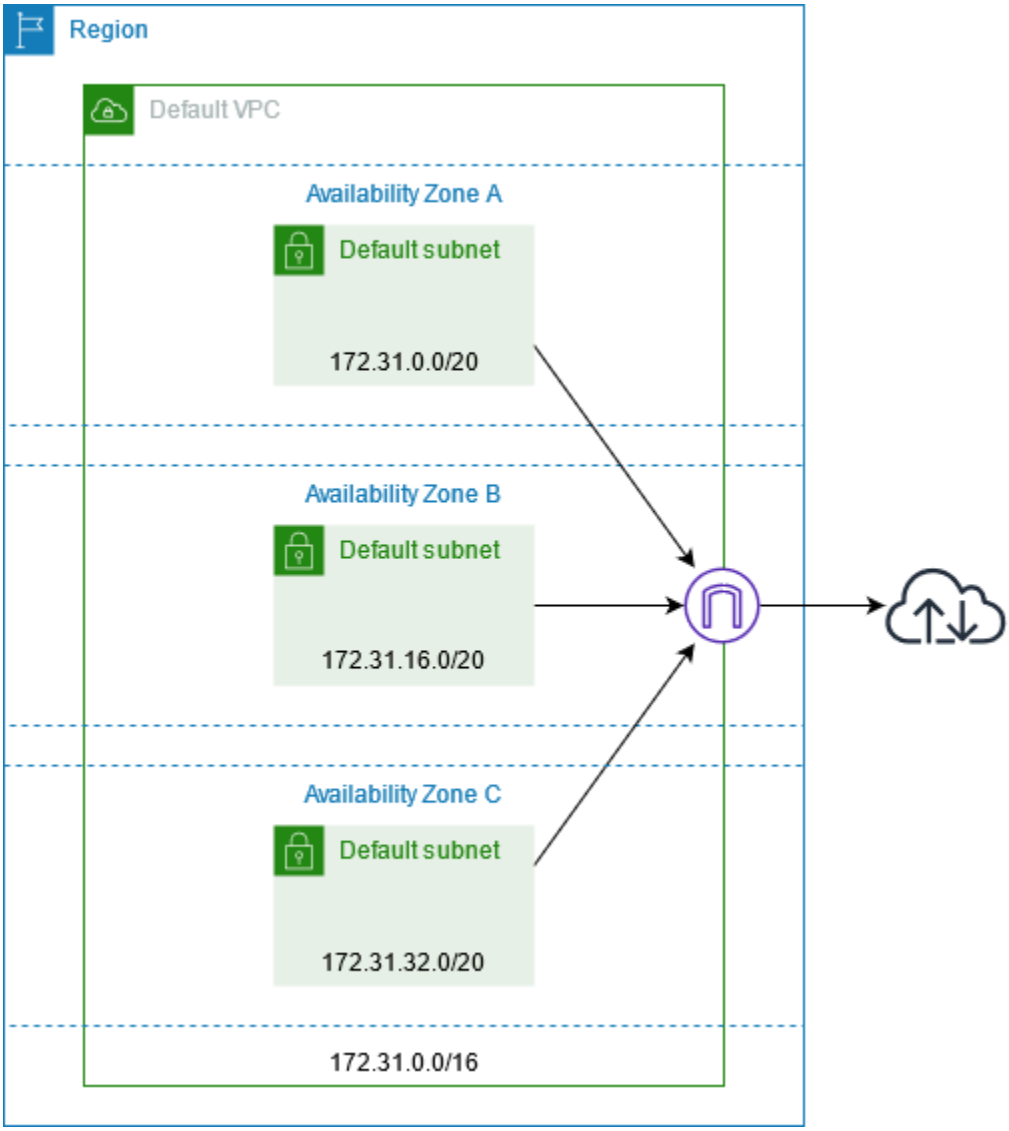
Daftar Isi


- VPC default Anda
- VPC nondefault

- Akses internet
- Subnet bersama
- IPv6-only subnet

## VPC default Anda

Saat Anda membuat AWS akun, kami membuat VPC default di setiap Wilayah. VPC default adalah VPC yang sudah dikonfigurasi dan siap untuk Anda gunakan. Misalnya, ada subnet default untuk setiap Availability Zone di setiap VPC default, gateway internet yang terpasang ke VPC, dan ada rute di tabel rute utama yang mengirimkan semua lalu lintas (0.0.0. 0/0) ke gateway internet. Anda dapat memodifikasi konfigurasi VPC default Anda sesuai kebutuhan. Misalnya, Anda dapat menambahkan subnet dan tabel rute.



## VPC nondefault

Alih-alih menggunakan VPC default untuk sumber daya Anda, Anda dapat membuat VPC Anda sendiri, seperti yang dijelaskan dalam Buat VPC di Panduan Pengguna Amazon VPC.

Berikut adalah beberapa hal yang perlu dipertimbangkan saat membuat VPC untuk instans EC2 Anda.

- Anda dapat menggunakan saran default untuk blok IPv4 CIDR atau memasukkan blok CIDR yang diperlukan oleh aplikasi atau jaringan Anda.
- Untuk memastikan ketersediaan tinggi, buat subnet di beberapa Availability Zone.
- Jika instans Anda harus dapat diakses dari internet, lakukan salah satu hal berikut:
  - Jika instans Anda dapat berada di subnet publik, tambahkan subnet publik. Biarkan kedua opsi DNS diaktifkan. Anda dapat menambahkan subnet privat secara opsional sekarang atau nanti.
  - Jika instans Anda harus berada di subnet pribadi, tambahkan hanya subnet pribadi. Anda dapat menambahkan gateway NAT untuk menyediakan akses internet ke instance di subnet pribadi. Jika instans Anda mengirim atau menerima volume lalu lintas yang signifikan di seluruh Availability Zone, buat gateway NAT di setiap Availability Zone. Jika tidak, Anda dapat membuat gateway NAT hanya di salah satu Availability Zones dan meluncurkan instance yang mengirim atau menerima lalu lintas lintas zona di Availability Zone yang sama dengan gateway NAT.

## Akses internet

Instans yang diluncurkan ke subnet default di VPC default memiliki akses ke internet, karena VPC default dikonfigurasi untuk menetapkan alamat IP publik dan nama host DNS, dan tabel rute utama dikonfigurasi dengan rute ke gateway internet yang dilampirkan ke VPC.

Untuk contoh yang Anda luncurkan di subnet nondefault dan VPC, Anda dapat menggunakan salah satu opsi berikut untuk memastikan bahwa instance yang Anda luncurkan di subnet ini memiliki akses ke internet:

- Konfigurasi gateway internet. Untuk informasi selengkapnya, lihat Hubungkan ke internet menggunakan gateway internet di Panduan Pengguna Amazon VPC.
- Konfigurasi gateway NAT publik. Untuk informasi selengkapnya, lihat Mengakses internet dari subnet privat di Panduan Pengguna Amazon VPC.

## Subnet bersama

Saat meluncurkan instans EC2 ke subnet VPC bersama, perhatikan hal berikut:

- Peserta dapat menjalankan instance di subnet bersama dengan menentukan ID subnet bersama. Peserta harus memiliki antarmuka jaringan apa pun yang mereka tentukan.
- Peserta dapat memulai, menghentikan, menghentikan, dan menjelaskan contoh yang telah mereka buat di subnet bersama. Peserta tidak dapat memulai, menghentikan, menghentikan, atau menjelaskan contoh yang dibuat oleh pemilik VPC di subnet bersama.
- Pemilik VPC tidak dapat memulai, menghentikan, menghentikan, atau menjelaskan instance yang dibuat oleh peserta dalam subnet bersama.
- Peserta dapat terhubung ke instans di subnet bersama menggunakan EC2 Instance Connect Endpoint. Peserta harus membuat EC2 Instance Connect Endpoint di subnet bersama. Peserta tidak dapat menggunakan Titik Akhir Instance Connect EC2 yang dibuat oleh pemilik VPC di subnet bersama.

Untuk informasi tentang sumber daya Amazon EC2 bersama, lihat berikut ini:

- the section called “Mengelola berbagi AMI dengan organisasi atau OU”
- the section called “Reservasi Kapasitas Bersama”
- the section called “Grup penempatan bersama”
- Cross-account Berbagi Host Khusus Amazon EC2

Untuk informasi selengkapnya tentang subnet bersama, lihat Bagikan VPC Anda dengan akun lain di Panduan Pengguna Amazon VPC.

## IPv6-only subnet

Instans EC2 yang diluncurkan di IPv6-only subnet menerima alamat IPv6 tetapi bukan alamat IPv4. Setiap instance yang Anda luncurkan ke IPv6-only subnet harus berupa Nitro-based instance.

# Jaringan Sekunder

Jaringan Sekunder adalah jaringan virtual yang dibuat khusus untuk kasus penggunaan jaringan khusus. Jaringan ini secara logis terisolasi di dalam partisi AWS cloud. Anda dapat membuat sumber

daya seperti subnet sekunder dalam jaringan sekunder Anda. Jaringan sekunder digabungkan erat dengan Amazon VPC, sehingga instans tertentu bersifat multi-homed dan diluncurkan ke VPC dan Jaringan Sekunder.

Jaringan Sekunder saat ini tersedia untuk jenis instans tertentu dan melalui reservasi kapasitas dengan komitmen jangka panjang. Hubungi tim akun Anda untuk informasi lebih lanjut jika menurut Anda jaringan sekunder mungkin bermanfaat bagi beban kerja Anda.

Daftar Isi


- Apa itu Jaringan Sekunder?
- Konsep Utama
- Arsitektur
- Pertimbangan tambahan
- Memulai
- Mengelola sumber daya Jaringan Sekunder
- Praktik terbaik desain jaringan
- Pemecahan masalah
- Kuota dan batas


## Apa itu Jaringan Sekunder?

Jaringan Sekunder menyediakan jaringan yang logis dan terisolasi, digunakan dalam kombinasi dengan jaringan VPC, sehingga instance multi-homed menjadi dua jaringan independen. Manfaat jaringan sekunder meliputi:

- High-performance jaringan untuk kasus penggunaan khusus dan protokol seperti konektivitas timur-barat untuk beban kerja ML
- Multi-tenant dukungan dengan isolasi logis
- Instans terintegrasi dengan mulus dengan VPC dan layanan AWS

## Konsep Utama

Jaringan Sekunder

Konstruksi jaringan regional yang menyediakan jaringan Layer 3 logis dengan blok CIDR IPv4 (mulai dari /28 hingga /12). Jaringan Sekunder beroperasi secara independen dari VPC pada infrastruktur jaringan yang dipartisi secara fisik.

Subnet Sekunder

Konstruksi Keter Zone-specific sediaan dalam Jaringan Sekunder, mirip dengan subnet VPC. Subnets sekunder mendukung blok CIDR mulai dari /28 hingga/12.

Antarmuka Sekunder

Antarmuka jaringan yang terpasang pada kartu jaringan sekunder, menyediakan konektivitas timur-barat dalam subnet sekunder. Antarmuka ini secara fisik dan logis terpisah dari Elastic Network Interfaces (ENI).

## Arsitektur

Instans EC2 yang mendukung Jaringan Sekunder bersifat multi-homed, artinya mereka dapat berkomunikasi dalam VPC dan Jaringan Sekunder secara bersamaan:

- VPC: Menyediakan TCP/IP konektivitas utara-selatan ke AWS layanan, penyimpanan, database, layanan jaringan, dan internet
- Jaringan Sekunder: Menyediakan konektivitas timur-barat antara instans khusus yang didukung

## Pertimbangan tambahan

- Antarmuka sekunder dikelola melalui RunInstances dan tidak dapat dibuat atau dihapus secara independen.
- Antarmuka sekunder tidak dapat attached/detached setelah instance diluncurkan.
- Alamat IP antarmuka sekunder tidak dapat diubah setelah diluncurkan.
- Fitur VPC seperti Grup Keamanan, NACL, Log Aliran tidak didukung di Jaringan Sekunder.

# Memulai

## Prasyarat

Sebelum meluncurkan instans dengan Jaringan Sekunder, pastikan Anda juga telah mengonfigurasi VPC di Wilayah yang ditargetkan dan Subnet di Zona Ketersediaan yang ditargetkan dari kapasitas EC2 Anda.

## Langkah 1: Buat Jaringan Sekunder

Buat Jaringan Sekunder di wilayah yang sama dengan VPC Anda. Ini adalah sumber daya regional yang menyediakan isolasi logis untuk lalu lintas RDMA Anda.

```
aws ec2 create-secondary-network \
  --network-type rdma \
  --ipv4-cidr-block 172.31.0.0/16 \
  --region us-east-2
```

Parameter:

- `--network-type`: Jenis jaringan (saat ini hanya rdma yang didukung)
- `--ipv4-cidr-block`: Blok IPv4 CIDR antara /28 dan /12
- `--region`: AWS Wilayah (US-East-2)

> **Note**
>
> Praktik Terbaik: Pilih blok CIDR yang tidak tumpang tindih dengan VPC CIDR Anda untuk menyederhanakan perutean di tingkat instance.

## Langkah 2: Buat Subnet Sekunder

Buat Subnet Sekunder di Zona Ketersediaan yang sama dengan subnet VPC Anda. Ini adalah AZ-specific sumber daya.

```
aws ec2 create-secondary-subnet \
  --secondary-network-id sn-1234567890abcdef0 \
  --ipv4-cidr-block 172.31.24.0/24 \
```

```
  --availability-zone us-east-2a
```

> **Note**
>
> Reservasi Alamat IP: Seperti subnet VPC, Amazon menyimpan 4 alamat IP pertama dan alamat IP terakhir di setiap Subnet Sekunder untuk penggunaan internal.

## Langkah 3: Luncurkan instans

Luncurkan instance ke subnet VPC dan Subnet Sekunder Anda. Instans akan multi-homed dengan konektivitas ke kedua jaringan.

```
aws ec2 run-instances \
  --image-id ami-12345678 \
  --count 1 \
  --instance-type <instance> \
  --key-name MyKeyPair \
  --instance-market-options '{"MarketType": "capacity-block"}' \
  --capacity-reservation-specification '{"CapacityReservationTarget": \
  {"CapacityReservationId": "cr-1234567890abcdef0"}}' \
  --network-interfaces \
    "NetworkCardIndex=0,DeviceIndex=0,Groups=sg-12345678,\
    SubnetId=subnet-0987654321fedcba0,InterfaceType=interface" \
  --secondary-interfaces \
    "NetworkCardIndex=1,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true", \
    "NetworkCardIndex=2,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true", \
    "NetworkCardIndex=3,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true", \
    "NetworkCardIndex=4,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true", \
    "NetworkCardIndex=5,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true", \
    "NetworkCardIndex=6,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true", \
    "NetworkCardIndex=7,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true", \
    "NetworkCardIndex=8,DeviceIndex=0,SecondarySubnetId=ss-98765421yxz,\
    InterfaceType=secondary,PrivateIpAddressCount=1,DeleteOnTermination=true"
```

Parameter Utama:

- `--network-interfaces`: Menentukan Nitro ENI utama untuk konektivitas VPC (Indeks Kartu Jaringan 0)
- `--secondary-interfaces`: Menentukan 8 antarmuka sekunder untuk konektivitas timur-barat dalam Subnet Sekunder (Indeks Kartu Jaringan 1-8)
- `InterfaceType=secondary`: Menunjukkan Antarmuka Sekunder

# Mengelola sumber daya Jaringan Sekunder

## Menjelaskan Jaringan Sekunder

Lihat detail tentang Jaringan Sekunder Anda:

```
aws ec2 describe-secondary-networks \
  --secondary-network-id sn-1234567890abcdef0
```

## Menjelaskan Subnets Sekunder

Lihat detail tentang Subnets Sekunder Anda:

```
aws ec2 describe-secondary-subnets \
  --secondary-subnet-id ss-98765421yxz
```

## Menjelaskan Antarmuka Sekunder

Lihat detail tentang Antarmuka Jaringan Sekunder yang dilampirkan ke instans Anda:

```
aws ec2 describe-secondary-interfaces \
  --filters "Name=attachment.instance-id,Values=i-1234567890abcdef0"
```

## Menghapus sumber daya

Hapus Subnet Sekunder:

```
aws ec2 delete-secondary-subnet \
  --secondary-subnet-id ss-98765421yxz
```

Hapus Jaringan Sekunder:

```
aws ec2 delete-secondary-network \
  --secondary-network-id sn-1234567890abcdef0
```

**Important**

Anda harus menghentikan semua instans dan menghapus semua Subnet Sekunder sebelum menghapus Jaringan Sekunder.

# Praktik terbaik desain jaringan

## Perencanaan CIDR

Hindari CIDR yang Tumpang  Tindih: Sementara Jaringan Sekunder secara fisik terisolasi dari VPC, menggunakan blok CIDR yang tidak tumpang tindih menyederhanakan konfigurasi routing di tingkat sistem operasi instance.

**Note**

Amazon menyimpan 5 alamat IP per subnet.

## Segregasi lalu lintas

Pisahkan berdasarkan Jaringan Sekunder: Buat Jaringan Sekunder terpisah untuk berbagai proyek, tim, atau batas keamanan. Jaringan Sekunder menyediakan isolasi logis antar instance. Instans tidak dapat berkomunikasi di Jaringan Sekunder yang berbeda.

Gunakan Beberapa Subnets: Dalam Jaringan Sekunder, gunakan beberapa Subnets Sekunder untuk mengelompokkan lalu lintas berdasarkan indeks GPU, Zona Ketersediaan, atau jenis beban kerja. Sebagai contoh, pola arsitektur umum adalah menyebarkan jaringan sekunder tunggal dengan 4 atau 8 subnet sekunder, di mana setiap subnet sekunder diselaraskan dengan sekelompok GPU indeks umum.

# Pemecahan masalah

## Kegagalan peluncuran instance

Masalah: Instance gagal diluncurkan dengan antarmuka Jaringan Sekunder.

Solusi:

- Pastikan AMI Anda menyertakan dukungan driver yang tepat
- Pastikan Subnet Sekunder Anda memiliki alamat IP yang cukup
- Konfirmasikan bahwa reservasi kapasitas Anda dalam keadaan “aktif”
- Periksa apakah Subnet Sekunder berada di Zona Ketersediaan yang sama dengan subnet VPC Anda

### Masalah konektivitas

Masalah: Tidak dapat membuat konektivitas RDMA antar instance.

Solusi:

- Verifikasi bahwa semua instans berada di Jaringan Sekunder dan Subnet Sekunder yang sama
- Periksa apakah driver antarmuka sekunder dimuat dengan benar pada instance
- Pastikan aplikasi Anda mengikat antarmuka jaringan yang benar
- Instans dalam subnet sekunder yang sama dapat dijangkau melalui rute langsung. Komunikasi lintas subnet tersedia melalui rute statis yang dijual melalui DHCP.

### Kesalahan API

Masalah: Panggilan API untuk operasi Jaringan Sekunder dan Subnet Sekunder gagal.

Solusi:

- Verifikasi izin IAM untuk ec2:CreateSecondaryNetwork, ec2:. CreateSecondarySubnet
- Periksa apakah blok CIDR berada dalam kisaran yang didukung (/28 hingga/12)
- Verifikasi bahwa Anda menggunakan Wilayah dan Zona Ketersediaan yang benar

## Kuota dan batas

Untuk meminta kenaikan kuota, gunakan Kuota AWS Layanan atau hubungi AWS Dukungan.

## Kuota dan batasan Jaringan Sekunder

| Sumber daya | Kuota | Dapat Disesuaikan |
| --- | --- | --- |
| Jaringan Sekunder per wilayah | 5 | Ya |
| Subnet Sekunder per Jaringan Sekunder | 200 | Ya |
| Ukuran blok CIDR | /28 ke /12 | Tidak |

# Keamanan dalam Amazon EC2

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama AWS antara Anda. Model tanggung jawab bersama menjelaskan hal ini sebagai keamanan cloud dan keamanan dalam cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Third-party auditor secara teratur menguji dan memverifikasi efektivitas keamanan kami sebagai bagian dari Program Kep AWS atuhan Program Kep . Untuk mempelajari tentang program kepatuhan yang berlaku untuk Amazon EC2, lihat Layanan dalam Lingkup AWS berdasarkan Layanan Program Kepatuhan dalam
- Keamanan dalam cloud – Tanggung jawab Anda meliputi area-area berikut:
  - Mengontrol akses jaringan pada instans Anda, misalnya, dengan mengonfigurasi VPC dan grup keamanan Anda. Untuk informasi selengkapnya, lihat Mengontrol lalu lintas jaringan.
  - Mengelola kredensial yang digunakan untuk terhubung ke instans Anda.
  - Mengelola sistem operasi tamu dan perangkat lunak yang diterapkan ke sistem operasi tamu, termasuk pembaruan dan patch keamanan. Untuk informasi selengkapnya, lihat Manajemen pembaruan untuk instans Amazon EC2.
  - Mengonfigurasi IAM role yang dilampirkan pada instans dan izin yang dikaitkan peran tersebut. Untuk informasi selengkapnya, lihat IAM role untuk Amazon EC2.

Dokumentasi ini akan membantu Anda dalam memahami cara menerapkan model tanggung jawab bersama saat Anda menggunakan Amazon EC2. Dokumentasi tersebut juga menunjukkan kepada Anda cara mengonfigurasi Amazon EC2 untuk memenuhi tujuan-tujuan keamanan dan kepatuhan Anda. Anda juga mempelajari cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan sumber daya Amazon EC2 Anda.

Daftar Isi


- Perlindungan data dalam Amazon EC2
- Keamanan infrastruktur di Amazon EC2
- Ketahanan dalam Amazon EC2

- Validasi kepatuhan untuk Amazon EC2
- Manajemen identitas dan akses untuk Amazon EC2
- Manajemen pembaruan untuk instans Amazon EC2
- Praktik terbaik keamanan untuk instans Windows
- Pasangan kunci Amazon EC2 dan instans Amazon EC2
- Grup keamanan Amazon EC2 untuk instans EC2 Anda
- NitroTPM untuk instans Amazon EC2
- Pengesahan instans Amazon EC2
- Credential Guard untuk instans Windows
- Mengakses Amazon EC2 menggunakan titik akhir VPC antarmuka

# Perlindungan data dalam Amazon EC2

Model tanggung jawab AWS bersama model tanggung jawab berlaku untuk perlindungan data di Amazon Elastic Compute Cloud. Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Anda juga bertanggung jawab atas tugas-tugas konfigurasi dan manajemen keamanan untuk Layanan AWS yang Anda gunakan. Untuk informasi selengkapnya tentang privasi data, lihat FAQ Privasi Data Ket . Untuk informasi tentang perlindungan data di Eropa, lihat Pusat Peraturan Umum Perlindungan Data (GDPR).

Untuk tujuan perlindungan data, kami menyarankan Anda melindungi Akun AWS kredenSIAL dan mengatur pengguna individu dengan AWS IAM Identity Center atau AWS Identity and Access Management (IAM). Dengan cara itu, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugasnya. Kami juga menyarankan supaya Anda mengamankan data dengan cara-cara berikut:

- Gunakan autentikasi multi-faktor (MFA) pada setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail. Untuk informasi tentang menggunakan CloudTrail jalur untuk menangkap AWS aktivitas, lihat Bekerja dengan CloudTrail jejak di Panduan AWS CloudTrail Pengguna.

- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default di dalamnya Layanan AWS.
- Gunakan layanan keamanan terkelola tingkat lanjut seperti Amazon Macie, yang membantu menemukan dan mengamankan data sensitif yang disimpan di Amazon S3.
- Jika Anda memerlukan modul kriptografi tervalidasi FIPS 140-3 saat mengakses AWS melalui antarmuka baris perintah atau API, gunakan titik akhir FIPS. Lihat informasi selengkapnya tentang titik akhir FIPS yang tersedia di Standar Pemrosesan Informasi Federal (FIPS) 140-3.

Kami sangat merekomendasikan agar Anda tidak pernah memasukkan informasi identifikasi yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam tanda atau bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan Amazon EC2 atau lainnya Layanan AWS menggunakan konsol, API AWS CLI, atau AWS SDK. Data apa pun yang Anda masukkan ke dalam tanda atau bidang isian bebas yang digunakan untuk nama dapat digunakan untuk log penagihan atau log diagnostik. Saat Anda memberikan URL ke server eksternal, kami sangat menganjurkan supaya Anda tidak menyertakan informasi kredensial di dalam URL untuk memvalidasi permintaan Anda ke server itu.

Daftar Isi


- Keamanan data Amazon EBS
- Enkripsi saat tidak aktif
- Enkripsi saat bergerak


## Keamanan data Amazon EBS

Volume Amazon EBS disajikan kepada Anda sebagai perangkat blok mentah yang tidak terformat. Perangkat-perangkat ini adalah perangkat logis yang dibuat pada infrastruktur EBS dan layanan Amazon EBS akan memastikan bahwa perangkat-perangkat tersebut secara logis kosong (yakni bahwa, blok mentah tersebut sudah dikosongkan atau mengandung data pseudorandom secara kriptografis) sebelum digunakan atau digunakan kembali oleh pelanggan.

Jika Anda memiliki prosedur yang mengharuskan semua data dihapus menggunakan metode tertentu, baik setelah atau sebelum digunakan (atau keduanya), seperti yang dirinci dalam DoD 5220.22-M (Manual Operasi Program Keamanan Industri Nasional) atau NIST 800-88(Pedoman untuk Sanitisasi Media), Anda memiliki kemampuan untuk melakukannya di Amazon EBS. Aktivitas tingkat blok tersebut akan tercermin ke media penyimpanan yang mendasarinya dalam layanan Amazon EBS tersebut.

# Enkripsi saat tidak aktif

Volume EBS

Enkripsi Amazon EBS adalah solusi enkripsi untuk volume dan snapshot EBS Anda. Ia menggunakan AWS KMS keys. Untuk informasi selengkapnya, lihat enkripsi Amazon EBS di Panduan Pengguna Amazon EBS.

[Instans Windows] Anda juga dapat menggunakan izin Microsoft EFS dan NTFS untuk enkripsi tingkat folder dan file.

Volume penyimpanan instans

Data pada volume penyimpanan instans NVMe dienkripsi menggunakan XTS-AES-256 cipher, diimplementasikan pada modul perangkat keras pada instance. Kunci yang digunakan untuk melakukan enkripsi pada data yang ditulis ke perangkat penyimpanan NVMe yang dilampirkan secara lokal berbeda-beda berdasarkan pelanggan dan berdasarkan volume. Kunci yang dihasilkan oleh, dan yang hanya berada di dalam, modul perangkat keras, yang tidak dapat diakses personil AWS . Kunci enkripsi tersebut akan dihancurkan saat instans dihentikan atau diakhiri dan tidak dapat dipulihkan. Anda tidak akan dapat menonaktifkan enkripsi ini dan Anda juga tidak dapat menyediakan kunci enkripsi Anda sendiri.

Data pada volume penyimpanan instans HDD pada instans H1, D3, dan D3en dienkripsi menggunakan kunci XTS-AES-256 satu kali.

Saat Anda menghentikan, melakukan hibernasi, atau mengakhiri instans, setiap blok penyimpanan dalam volume penyimpanan instans akan diatur ulang. Oleh karena itu, data Anda tidak dapat diakses melalui penyimpanan instans dari instans yang lain.

Memori

Enkripsi memori diaktifkan pada instans-instans berikut:

- Instans dengan AWS Graviton2 atau prosesor AWS Graviton yang lebih baru mendukung enkripsi memori yang selalu aktif. Kunci enkripsi yang secara aman dihasilkan dalam sistem host, tidak meninggalkan sistem host, dan akan hancur ketika host tersebut di-reboot atau dimatikan. Untuk informasi lainnya, lihat Prosesor AWS Graviton.
- Instans dengan prosesor Intel Xeon Scalable generasi ke-3 (Ice Lake), seperti instans M6i, dan prosesor Intel Xeon Scalable generasi ke-4 (Sapphire Rapids), seperti instans M7i. Prosesor-

prosesor ini mendukung enkripsi memori yang selalu aktif menggunakan Intel Total Memory Encryption (TME).

- Instans dengan prosesor AMD EPYC generasi ke-3 (Milan), seperti instans M6a, dan prosesor AMD EPYC generasi ke-4 (Genoa), seperti instans M7a. Prosesor ini mendukung enkripsi memori yang selalu aktif menggunakan AMD Secure Memory Encryption (SME).
- AMD Secure Encry Virtualization-Secure pted Nested Paging (SEV-SNP) didukung untuk beberapa jenis AMD-based instans. Untuk informasi selengkapnya, lihat Temukan jenis instans EC2 yang mendukung AMD SEV-SNP.

## Enkripsi saat bergerak

Enkripsi pada lapisan fisik

Semua data yang mengalir mel AWS intasi Wilayah AWS melalui jaringan global secara otomatis dienkripsi di lapisan fisik sebelum meninggalkan fasilitas AWS yang aman. Semua lalu lintas antara AZ akan dienkripsi. Lapisan-lapisan enkripsi tambahan, termasuk yang tercantum dalam bagian ini, dapat memberikan perlindungan tambahan.

Enkripsi yang disediakan oleh peering Amazon VPC dan peering lintas wilayah Transit Gateway

Semua lalu lintas lintas wilayah yang menggunakan peering Amazon VPC dan peering Transit Gateway secara otomatis dienkripsi massal saat keluar dari Wilayah. Lapisan tambahan enkripsi secara otomatis disediakan di lapisan fisik untuk semua lalu lintas sebelum meninggalkan fasilitas yang AWS aman, seperti yang disebutkan sebelumnya di bagian ini.

Enkripsi antar instans

AWS menyediakan konektivitas aman dan pribadi antara instans EC2 dari semua jenis. Selain itu, beberapa tipe instans menggunakan kemampuan offload dari perangkat keras Nitro System yang mendasarinya untuk secara otomatis mengenkripsi lalu lintas dalam transit antar instans. Enkripsi ini menggunakan algoritma Authenticated Encryption with Associated Data (AEAD), dengan enkripsi 256-bit. Tidak ada dampak terhadap performa jaringan. Untuk mendukung enkripsi lalu lintas dalam transit tambahan ini antara instans, persyaratan-persyaratan berikut harus dipenuhi:

- Instans-instans tersebut menggunakan tipe instans berikut:
  - Tujuan umum: M5dn, M5n, M5Zn, M6a, M6i, M6id, M6idn, M6in, M7a, M7g, M7gd, M7i,, M8a, M8azN, M8g, M8gD, M8gD, M8Gn, M8i, M8id,, M8in, M8in, M8iD N, M8in, M8ib M7i-flex, M8idb, M9g, M9gd,, M8i-flex Mac-m4 Mac-m4pro

  - Komputasi dioptimalkan:  C5n, C6a, C6gn, C6i, C6id, C6in, C7a, C7g, C7gD, C7gn, C7i,, C8a, C8g, C8g, C8gD, C8gn, C8i, C8id,, C8in, C8ine, C8iB, C9g C7i-flex, C9gd C8i-flex
  - Memori dioptimalkan:  R5dn, R5n, R6a, R6i, R6id, R6idn, R6in, R7a, R7g, R7gd, R7i, R7iz, R8a, R8g, R8g, R8Gd, R8gn, R8i, R8id,, R8in, R8idn, R8ib, R8idb, R9g, R9gd,,,,,,,,,,, X2IDn, x2iEDN, x2iEZn, x8g, x8AEDz, X8i R8i-flex U-3tb1 U-6tb1 U-9tb1 U-12tb1 U-18tb1 U-24tb1 U7i-6tb U7i-8tb U7i-12tb U7in-16tb U7in-24tb U7in-32tb U7inh-32tb
  - Penyimpanan dioptimalkan:  D3, D3en, i3en, i4G, i4i, i7i, i7ie, i8G, i8GE, iM4GN, IS4gen
  - Komputasi yang dipercepat:  DL1, DL2q, F2, G4ad, G4dn, G5, G6, G6e, G6f, Gr6, Gr6f, G7, G7e, Inf1, Inf2, P3dN, P4d, P4de, P5, P5e, P5en,,,, Trn1, Trn1n, Trn2, Trn2u, VT1 P6-B200 P6-B300 P6e-GB200
  - High-performance komputasi:  hpc6a, hpc6id, hpc7a, hpc7g, hpc8a
- Instans-instans tersebut berada dalam Wilayah yang sama.
- Instans-instans tersebut berada dalam VPC yang sama atau VPC yang di-peering yang sama, dan lalu lintas tidak melewati perangkat atau layanan jaringan virtual, seperti penyeimbang beban atau gateway transit.

Lapisan tambahan enkripsi secara otomatis disediakan di lapisan fisik untuk semua lalu lintas sebelum meninggalkan fasilitas yang AWS aman, seperti yang disebutkan sebelumnya di bagian ini.

Untuk melihat jenis instans yang mengenkripsi lalu lintas dalam transit antar instans menggunakan AWS CLI

Gunakan perintah describe-instance-types berikut ini.

```
aws ec2 describe-instance-types \
    --filters Name=network-info.encryption-in-transit-supported,Values=true \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

Enkripsi ke dan dari AWS Outposts

Outpost membuat koneksi jaringan khusus yang disebut tautan  layanan  ke Wilayah as AWS alnya dan, secara opsional, konektivitas pribadi ke subnet VPC yang Anda tentukan. Semua lalu lintas yang melalui koneksi tersebut sudah sepenuhnya dienkripsi. Untuk informasi selengkapnya, lihat Konektivitas melalui tautan layanan dan Enkripsi dalam transit di Panduan Pengguna AWS Outposts .

Enkripsi akses jarak jauh

Protokol SSH dan RDP menyediakan saluran komunikasi yang aman untuk akses jarak jauh ke instans Anda, baik secara langsung maupun melalui EC2 Instance Connect. Akses jarak jauh ke instans Anda menggunakan AWS Systems Manager Pengelola Sesi atau Perintah Jalankan dienkripsi menggunakan TLS 1.2, dan permintaan untuk membuat koneksi ditandatangani menggunakan SIGv4, dan diautentikasi serta disahkan oleh. AWS Identity and Access Management

Anda bertanggung jawab untuk menggunakan protokol enkripsi, seperti Keamanan Lapisan Pengangkutan (TLS), untuk mengenkripsi data bergerak sensitif Anda yang bergerak antara klien dan instans Amazon EC2 Anda.

(Instans Windows) Pastikan untuk hanya mengizinkan koneksi terenkripsi antara instans EC2 dan titik akhir AWS API atau layanan jaringan jarak jauh sensitif lainnya. Anda dapat menerapkan hal ini melalui grup keamanan ke luar atau aturan Windows Firewall.

# Keamanan infrastruktur di Amazon EC2

Sebagai layanan terkelola, Amazon Elastic Compute Cloud dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat AWS Keamanan Cloud. Untuk merancang AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat Perlindungan Infra struktur di Security Pillar AWS Well-Architected Framework.

Anda menggunakan panggilan API yang di AWS publikasikan untuk mengakses Amazon EC2 melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Pengangkutan (TLS). Kami mensyaratkan TLS 1.2 dan menganjurkan TLS 1.3.
- Suite Cipher dengan kerahasiaan maju sempurna (PFS) seperti DHE (Ephemeral) atau ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Diffie-Hellman Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Untuk informasi selengkapnya, lihat Perlindungan Infrastruktur di Pil ar Keamanan — Ker AWS Well-Architected angka Kerja.

## Isolasi jaringan

Virtual Private Cloud (VPC) adalah jaringan virtual di area Anda sendiri yang terisolasi secara logis di AWS Cloud. Gunakan VPC yang terpisah untuk melakukan isolasi terhadap infrastruktur berdasarkan beban kerja atau entitas organisasi.

subnet adalah serangkaian alamat IP di VPC. Saat Anda meluncurkan instans, Anda meluncurkan instans tersebut ke dalam subnet dalam VPC Anda. Gunakan subnet untuk melakukan isolasi terhadap jenjang-jenjang aplikasi Anda (misalnya web, aplikasi, dan basis data) dalam satu VPC. Gunakan subnet privat untuk instans Anda jika instan tersebut tidak dapat diakses secara langsung dari internet.

Untuk memanggil API Amazon EC2 dari VPC menggunakan alamat IP privat, gunakan AWS PrivateLink. Untuk informasi selengkapnya, lihat Mengakses Amazon EC2 menggunakan titik akhir VPC antarmuka.

## Isolasi pada host fisik

Instans-instans EC2 yang beragam pada host fisik yang sama diisolasi satu sama lain seolah-olah mereka berada dalam host fisik terpisah. Hypervisor mengisolasi CPU dan memori, dan instans disediakan dalam bentuk disk virtual, bukan berupa akses ke perangkat disk mentah.

Saat Anda menghentikan atau mengakhiri instans, memori yang dialokasikan untuk instans itu dibersihkan (diatur ke nol) oleh hypervisor sebelum dialokasikan ke instans baru, dan setiap blok penyimpanan akan diatur ulang. Hal ini untuk memastikan agar data Anda tidak terekspos secara tidak sengaja ke instans lain.

Alamat MAC jaringan secara dinamis ditetapkan ke instans oleh infrastruktur AWS jaringan. Alamat IP dapat ditetapkan secara dinamis ke instans oleh infrastruktur jaringan AWS , atau ditetapkan oleh administrator EC2 melalui permintaan API terautentikasi. AWS Jaringan memungkinkan instance untuk mengirim lalu lintas hanya dari alamat MAC dan IP yang ditetapkan untuk mereka. Jika tidak, lalu lintas akan menurun.

Secara default, instans tidak dapat menerima lalu lintas yang tidak secara khusus ditujukan padanya. Jika Anda perlu menjalankan layanan terjemahan alamat jaringan (NAT), routing, atau firewall pada instans Anda, Anda dapat menonaktifkan source/destination pemeriksaan antarmuka jaringan.

## Mengontrol lalu lintas jaringan

Pertimbangkan opsi-opsi berikut untuk mengontrol lalu lintas jaringan pada instans EC2 Anda:

- Batasi akses ke instans Anda menggunakan grup keamanan. Konfigurasikan aturan yang memungkinkan lalu lintas jaringan minimum yang diperlukan. Misalnya, Anda dapat mengizinkan lalu lintas hanya dari rentang alamat untuk jaringan perusahaan Anda atau hanya untuk protokol tertentu, seperti HTTPS. Untuk instans Windows, izinkan lalu lintas manajemen Windows dan koneksi keluar minimal.

- Manfaatkan grup keamanan sebagai mekanisme utama untuk mengontrol akses jaringan ke instans Amazon EC2. Jika perlu, gunakan ACL jaringan secara terbatas untuk menyediakan kontrol jaringan stateless dan secara garis besar. Grup keamanan bersifat lebih serba guna daripada ACL jaringan karena kemampuannya untuk melakukan pemfilteran paket stateful dan membuat aturan yang mengacu pada grup keamanan lainnya. Namun demikian, ACL jaringan akan efektif sebagai pengendali sekunder untuk menolak subset lalu lintas tertentu atau untuk menyediakan pagar pengaman subnet tingkat tinggi. Selain itu, karena ACL jaringan berlaku untuk keseluruhan subnet, ACL jaringan ini dapat digunakan sebagai pertahanan mendalam jika instans diluncurkan secara tidak sengaja tanpa grup keamanan yang tepat.
- [Instans Windows] Kelola pengaturan Windows Firewall secara terpusat dengan Objek Kebijakan Grup (GPO) untuk lebih meningkatkan kontrol jaringan. Para pelanggan sering menggunakan Windows Firewall untuk mendapatkan visibilitas ke dalam lalu lintas jaringan lebih jauh dan untuk melengkapi filter grup keamanan, membuat aturan-aturan lanjutan untuk memblokir aplikasi tertentu agar tidak mengakses jaringan atau untuk memfilter lalu lintas dari alamat IP subset. Sebagai contoh, Windows Firewall dapat membatasi akses ke alamat IP layanan metadata EC2 untuk pengguna atau aplikasi tertentu. Atau, layanan yang dapat diakses publik dapat menggunakan grup keamanan untuk membatasi lalu lintas ke port tertentu dan menggunakan Windows Firewall untuk memelihara daftar alamat IP yang diblokir secara eksplisit.
- Gunakan subnet privat untuk instans Anda jika instan tersebut tidak dapat diakses secara langsung dari internet. Gunakan host bastion atau gateway NAT untuk akses internet dari instans di subnet privat.
- [Instans Windows] Gunakan protokol administrasi aman seperti enkapsulasi RDP over. SSL/TLS Mulai Cepat Gateway Desktop Jarak Jauh memberikan praktik terbaik untuk menerapkan gateway desktop jarak jauh, termasuk mengonfigurasi RDP untuk digunakan. SSL/TLS
- [Instans Windows] Gunakan Active Directory atau Directory Service untuk mengontrol dan memantau pengguna dan grup interaktif akses pengguna dan grup ke instans Windows secara ketat dan terpusat, dan hindari izin pengguna lokal. Selain itu, hindari penggunaan Domain Administrator dan buatlah lebih banyak akun berbasis peran yang terperinci dan spesifik untuk aplikasi. Just Enough Administration (JEA) mengizinkan perubahan-perubahan pada instans Windows dikelola tanpa akses interaktif atau administrator. Selain itu, JEA memungkinkan organisasi untuk mengunci akses administratif pada subset perintah Windows PowerShell yang diperlukan untuk administrasi instans. Untuk informasi tambahan, lihat Prakti k Terbaik untuk Keamanan, Identitas, & Kepatuhan.
- [Instans Windows] Administrator sistem harus menggunakan akun Windows dengan akses terbatas untuk melakukan aktivitas sehari-hari, dan hanya meningkatkan akses bila diperlukan untuk

melakukan perubahan konfigurasi tertentu. Selain itu, akses instans Windows secara langsung hanya bila benar-benar diperlukan. Sebaliknya, manfaatkan sistem manajemen konfigurasi pusat seperti EC2 Run Command, Systems Center Configuration Manager (SCCM), Windows PowerShell DSC, atau Amazon EC2 Systems Manager (SSM) untuk mendorong perubahan pada peladen Windows.

- Lakukan konfigurasi pada tabel rute subnet Amazon VPC dengan rute jaringan syarat minimal. Misalnya, tempatkan hanya instans Amazon EC2 yang memerlukan akses Internet langsung ke subnet dengan rute ke gateway internet, dan tempatkan hanya instans Amazon EC2 yang memerlukan akses langsung ke jaringan internal ke subnet dengan rute ke gateway pribadi virtual.
- Pertimbangkan untuk menggunakan grup keamanan tambahan atau antarmuka jaringan untuk mengontrol dan meng-audit lalu lintas pengelolaan instans Amazon EC2 secara terpisah dari lalu lintas aplikasi reguler. Pendekatan ini memungkinkan pelanggan untuk melaksanakan kebijakan IAM khusus untuk kontrol pengubahan, mempermudahnya untuk meng-audit perubahan pada aturan grup keamanan atau skrip verifikasi aturan otomatis. Menggunakan beberapa antarmuka jaringan juga menyediakan opsi tambahan untuk mengendalikan lalu lintas jaringan, termasuk kemampuan untuk membuat kebijakan routing berbasis host atau memanfaatkan aturan routing subnet VPC yang berbeda berdasarkan subnet yang ditetapkan dari antarmuka jaringan.
- Gunakan AWS Virtual Private Network atau Direct Connect untuk membuat koneksi pribadi dari jaringan jarak jauh Anda ke VPC Anda. Untuk informasi selengkapnya, lihat Network-to-Amazon Opsi Konektivitas VPC.
- Gunakan Log Aliran VPC untuk memantau lalu lintas yang menjangkau instans Anda.
- Gunakan Per GuardDuty lindungan Malware untuk mengidentifikasi perilaku mencurigakan yang mengindikasikan perangkat lunak berbahaya pada instans Anda yang dapat membahayakan beban kerja Anda, menggunakan kembali sumber daya untuk penggunaan berbahaya, dan mendapatkan akses tidak sah ke data Anda.
- Gunakan GuardDuty Pemantauan Runtime untuk mengidentifikasi dan menanggapi potensi ancaman terhadap instans Anda. Untuk informasi selengkapnya, lihat Cara Runtime Monitoring bekerja dengan inst ans Amazon EC2.
- Gunakan AWS Security Hub CSPM, Reachability Analyzer, atau Network Access Analyzer untuk memeriksa aksesibilitas jaringan yang tidak diinginkan dari instans Anda.
- Gunakan EC2 Instance Connect untuk terhubung ke instans Anda menggunakan Secure Shell (SSH) tanpa perlu berbagi dan mengelola kunci SSH.
- Gunakan Manaj AWS Systems Manager er Sesi untuk mengakses instans Anda dari jarak jauh alih-alih membuka port SSH atau RDP masuk dan mengelola pasangan kunci.

- Gunakan AWS Systems Manager Run Command untuk mengotomatiskan tugas administratif umum alih-alih menghubungkan ke instans Anda.
- [Instans Windows] Banyak peran OS Windows dan aplikasi bisnis Microsoft juga menyediakan fungsionalitas yang ditingkatkan seperti pembatasan Rentang Alamat IP dalam IIS, kebijakan pem TCP/IP filteran di Microsoft SQL Server, dan kebijakan filter koneksi di Microsoft Exchange. Fungsionalitas pembatasan jaringan dalam lapisan aplikasi dapat menyediakan lapisan pertahanan tambahan untuk server aplikasi bisnis penting.

Amazon VPC mendukung kontrol keamanan jaringan tambahan, seperti gateway, server proxy, dan opsi pemantauan jaringan. Untuk informasi selengkapnya, lihat Mengontrol lalu lintas jaringan di Panduan Pengguna Amazon VPC.

# Ketahanan dalam Amazon EC2

Infrastruktur AWS global dibangun di sekitar AWS Wilayah dan Zona Ketersediaan. Wilayah memberikan beberapa Zona Ketersediaan yang terpisah dan terisolasi secara fisik, yang terkoneksi melalui jaringan latensi rendah, throughput tinggi, dan sangat redundan. Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan fail over di antara zona tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur pusat data tunggal atau multi tradisional.

Jika Anda harus melakukan replikasi data atau aplikasi Anda pada jarak geografis yang lebih luas, gunakan Zona Lokal AWS . Zona AWS Lokal adalah perpanjangan dari AWS Wilayah dalam kedekatan geografis dengan pengguna Anda. Zona Lokal memiliki koneksinya sendiri ke internet dan mendukung Direct Connect. Seperti semua AWS Wilayah, AWS Local Zones benar-benar terisolasi dari AWS Zona lain.

Jika Anda perlu mereplikasi data atau aplikasi Anda di Zona AWS Lokal, AWS sarankan Anda menggunakan salah satu zona berikut sebagai zona failover:

- Zona Lokal Lainnya
- Zona Ketersediaan dalam Wilayah yang bukan merupakan zona induk. Anda dapat menggunakan describe-availability-zonesperintah untuk melihat zona induk.

Untuk informasi selengkapnya tentang AWS Wilayah dan Availability Zone, lihat Infrastruktur AWS Global.

Selain infrastruktur AWS global, Amazon EC2 menawarkan fitur-fitur berikut untuk mendukung ketahanan data Anda:

- Menyalin AMIs lintas Wilayah
- Menyalin snapshot EBS di seluruh Wilayah
- Mengotomatiskan EBS yang didukung AMIs menggunakan Amazon Data Lifecycle Manager
- Menerapkan otomatisasi snapshot EBS menggunakan Amazon Data Lifecycle Manager
- Menjaga kondisi dan ketersediaan armada Anda menggunakan Amazon EC2 Auto Scaling
- Mendistribusikan lalu lintas masuk pada berbagai instans dalam satu Zona Ketersediaan atau beberapa Zona Ketersediaan menggunakan Elastic Load Balancing

## Validasi kepatuhan untuk Amazon EC2

Pengalaman AWS pendaftaran baru kami tidak dirancang untuk beban kerja yang diatur. Jika Anda menggunakan pengalaman AWS pendaftaran baru kami, tetapi ingin menggunakannya AWS untuk beban kerja yang diatur, Anda dapat mendaftar untuk AWS (lanjutan) atau mengaktifkan fitur lanjutan untuk lingkungan Anda AWS .

Untuk mempelajari apakah an Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat Layanan AWS di Lingkup berdasarkan Program Kepatuhan dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat Program Kep AWS atuhan Program Jaminan .

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat Mengunduh Laporan di AWS Artifact .

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. Untuk informasi selengkapnya tentang tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS, lihat Dokumentasi AWS Keamanan.

## Manajemen identitas dan akses untuk Amazon EC2

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. Administrator IAM mengontrol siapa yang dapat diautentikasi (masuk) dan di otorisasi (memiliki izin) untuk menggunakan sumber daya Amazon EC2. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

KredenSIAL keamanan mengidentifikasi Anda ke layanan di AWS dan memberi Anda akses ke AWS sumber daya, seperti sumber daya Amazon EC2 Anda. Anda dapat menggunakan fitur Amazon EC2 dan IAM untuk mengizinkan pengguna, layanan, dan aplikasi lain menggunakan sumber daya Amazon EC2 tanpa membagikan kredenSIAL keamanan Anda. Anda dapat menggunakan IAM untuk mengontrol cara pengguna lain menggunakan sumber daya di Anda Akun AWS, dan Anda dapat menggunakan grup keamanan untuk mengontrol akses ke instans Amazon EC2 Anda. Anda dapat memilih untuk mengizinkan penggunaan penuh atau terbatas sumber daya Amazon EC2 Anda.

Jika Anda seorang pengembang, Anda dapat menggunakan peran IAM untuk mengelola kredenSIAL keamanan yang dibutuhkan oleh aplikasi yang Anda jalankan pada instans EC2. Setelah Anda melampirkan peran IAM ke instans, aplikasi yang berjalan pada instans dapat mengambil kredenSIAL dari Layanan Metadata Instance (IMDS).

Untuk praktik terbaik untuk mengamankan AWS sumber daya Anda menggunakan IAM, lihat Prakti k terbaik keamanan di IAM di Panduan Pengguna IAM.

Daftar Isi


- Identity-based kebijakan untuk Amazon EC2
- Contoh kebijakan untuk mengontrol akses Amazon EC2 API
- Contoh kebijakan untuk mengontrol akses ke konsol Amazon EC2
- AWS kebijakan terkelola untuk Amazon EC2
- IAM role untuk Amazon EC2


## Identity-based kebijakan untuk Amazon EC2

Secara default, pengguna tidak memiliki izin untuk membuat atau memodifikasi sumber daya Amazon EC2, atau melakukan tugas menggunakan API Amazon EC2, konsol Amazon EC2, atau CLI. Untuk memungkinkan pengguna membuat atau memodifikasi sumber daya dan melakukan tugas, Anda harus membuat kebijakan IAM yang memberikan izin kepada pengguna untuk menggunakan sumber daya serta tindakan API tertentu yang akan mereka perlukan, lalu melampirkan kebijakan tersebut ke pengguna, grup, atau peran IAM yang memerlukan izin tersebut.

Saat Anda melampirkan kebijakan ke pengguna atau grup pengguna atau peran, kebijakan tersebut akan mengizinkan atau menolak izin pengguna untuk melakukan tugas tertentu pada sumber daya tertentu. Untuk informasi umum selengkapnya tentang kebijakan IAM, lihat Kebijakan dan izin di IAM di Panduan Pengguna IAM. Untuk informasi selengkapnya tentang mengelola dan membuat kebijakan IAM, lihat Mengel ola kebijakan IAM.

kebijakan IAM harus mengizinkan atau menolak izin untuk menggunakan satu atau beberapa tindakan Amazon EC2. Kebijakan tersebut juga harus menentukan sumber daya yang dapat digunakan bersama dengan tindakan tersebut, yang dapat berupa semua sumber daya, atau dalam beberapa kasus, sumber daya tertentu. Kebijakan ini juga dapat mencakup syarat-syarat yang Anda terapkan pada sumber daya.

Untuk memulai, Anda dapat memeriksa apakah kebijakan terkel AWS ola untuk Amazon EC2 memenuhi kebutuhan Anda. Jika tidak, Anda dapat membuat kebijakan khusus Anda sendiri. Untuk informasi selengkapnya, lihat the section called “AWS kebijakan yang dikelola”.

Daftar Isi


- Sintaksis kebijakan
- Tindakan-tindakan untuk Amazon EC2
- Izin tingkat sumber daya yang mendukung tindakan API Amazon EC2
- Amazon Resource Name (ARN) untuk Amazon EC2
- Kunci syarat untuk Amazon EC2
- Kontrol akses menggunakan akses berbasis atribut
- Berikan izin kepada pengguna, grup, dan peran
- Memeriksa apakah pengguna memiliki izin yang diperlukan


## Sintaksis kebijakan

kebijakan IAM adalah dokumen JSON yang terdiri dari satu atau beberapa pernyataan. Masing-masing pernyataan memiliki struktur sebagai berikut.

```
{
  "Statement":[{
    "Effect":"effect",
    "Action":"action",
    "Resource":"arn",
    "Condition":{
      "condition":{
        "key":"value"
        }
      }
    }
  ]
```

```
}
```

Ada berbagai elemen yang membentuk pernyataan:

- Efek: Efek bisa berupa `Allow` atau `Deny`. Secara default, pengguna tidak memiliki izin untuk menggunakan sumber daya dan tindakan API, sehingga semua permintaan akan ditolak. izin eksplisit akan menggantikan izin default. penolakan eksplisit akan menggantikan izin apa pun.
- Tindakan: Tindakan adalah tindakan API tertentu yang Anda izinkan atau tolak. Untuk mempelajari tentang cara menentukan tindakan, lihat Tindakan-tindakan untuk Amazon EC2.
- Sumber daya: Sumber daya yang dipengaruhi oleh tindakan. Beberapa tindakan Amazon EC2 API memungkinkan Anda untuk menyertakan sumber daya tertentu dalam kebijakan Anda yang dapat dibuat atau dimodifikasi oleh tindakan tersebut. Anda dapat menentukan sumber daya menggunakan Amazon Resource Name (ARN) atau menggunakan wildcard (*) untuk menunjukkan bahwa pernyataan berlaku untuk semua sumber daya. Untuk informasi selengkapnya, lihat Izin tingkat sumber daya yang mendukung tindakan API Amazon EC2.
- Syarat: Syarat-syarat bersifat opsional. Syarat-syarat ini dapat digunakan untuk mengendalikan kapan kebijakan Anda berlaku. Untuk informasi selengkapnya tentang cara menentukan syarat untuk Amazon EC2, lihat Kunci syarat untuk Amazon EC2.

Untuk informasi selengkapnya tentang persyaratan kebijakan, lihat Referensi kebijakan IAM JSON di Panduan Pengguna IAM. Misalnya pernyataan kebijakan IAM untuk Amazon EC2, lihat Contoh kebijakan untuk mengontrol akses Amazon EC2 API.

## Tindakan-tindakan untuk Amazon EC2

Dalam pernyataan kebijakan IAM, Anda dapat menentukan tindakan API apa pun dari layanan apa pun yang mendukung IAM. Untuk Amazon EC2, gunakan awalan berikut ini dengan nama dari tindakan API: `ec2:`. Misalnya: `ec2:RunInstances` dan `ec2:CreateImage`.

Untuk menetapkan beberapa tindakan dalam satu pernyataan, pisahkan tindakan-tindakan tersebut menggunakan koma seperti berikut:

```
"Action": ["ec2:action1", "ec2:action2"]
```

Anda juga dapat menentukan beberapa tindakan menggunakan wildcard. Sebagai contoh, Anda dapat menentukan semua tindakan yang namanya dimulai dengan kata “Describe” seperti berikut ini:

```
"Action": "ec2:Describe*"
```

> **Note**
>
> Saat ini, tindakan API Describe* Amazon EC2 tidak mendukung izin tingkat sumber daya. Untuk informasi selengkapnya tentang izin tingkat sumber daya untuk Amazon EC2, lihat Identity-based kebijakan untuk Amazon EC2.

Untuk menentukan semua tindakan API Amazon EC2, gunakan wildcard * sebagai berikut:

```
"Action": "ec2:*"
```

Untuk melihat daftar tindakan Amazon EC2, lihat Actions defined by Amazon EC2 (Tindakan yang ditentukan oleh Amazon EC2) di Referensi Otorisasi Layanan.

### Izin tingkat sumber daya yang mendukung tindakan API Amazon EC2

Resource-level izin  mengacu pada kemampuan untuk menentukan sumber daya mana pengguna diizinkan untuk melakukan tindakan. Amazon EC2 memiliki dukungan parsial untuk izin tingkat sumber daya. Artinya untuk tindakan Amazon EC2 tertentu, Anda dapat mengontrol kapan para pengguna diizinkan untuk menggunakan tindakan tersebut berdasarkan syarat yang harus dipenuhi, atau sumber daya tertentu yang diizinkan untuk digunakan oleh pengguna. Sebagai contoh, Anda dapat memberikan izin kepada para pengguna untuk meluncurkan instans, tetapi hanya untuk tipe instans tertentu, dan hanya menggunakan AMI tertentu.

Untuk menentukan sumber daya di pernyataan kebijakan IAM, gunakan Amazon Resource Name (ARN) sumber daya tersebut. Untuk informasi selengkapnya tentang cara menentukan nilai ARN, lihat Amazon Resource Name (ARN) untuk Amazon EC2. Jika tindakan API tidak mendukung ARN individu, Anda harus menggunakan wildcard (*) untuk menentukan bahwa semua sumber daya dapat dipengaruhi oleh tindakan tersebut.

Untuk melihat tabel yang mengidentifikasi tindakan API Amazon EC2 mana yang mendukung izin tingkat sumber daya, dan ARN serta kunci syarat yang dapat Anda gunakan di kebijakan, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2.

Ingatlah bahwa Anda dapat menerapkan izin tingkat sumber daya berbasis tanda dalam kebijakan IAM yang Anda gunakan untuk tindakan API Amazon EC2. Hal ini akan memberikan Anda kontrol yang lebih baik atas sumber daya yang dapat dibuat, dimodifikasi, atau digunakan oleh seorang

pengguna. Untuk informasi selengkapnya, lihat Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan.

## Amazon Resource Name (ARN) untuk Amazon EC2

Setiap pernyataan kebijakan IAM berlaku untuk sumber daya yang Anda tentukan menggunakan ARN.

ARN memiliki sintaksis umum sebagai berikut:

```
arn:aws:[service]:[region]:[account-id]:resourceType/resourcePath
```

layanan

Layanan (contohnya, `ec2`).

wilayah

Wilayah untuk sumber daya (contohnya, `us-east-1`).

account-id

ID AWS akun, tanpa tanda hubung (misalnya,`123456789012`).

resourceType

Jenis dari sumber daya (contohnya, `instance`).

resourcePath

jalur yang mengidentifikasi sumber daya. Anda dapat menggunakan wildcard * dalam jalur Anda.

Sebagai contoh, Anda dapat mengindikasikan instans tertentu (`i-1234567890abcdef0`) dalam pernyataan Anda menggunakan ARN seperti berikut ini.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0"
```

Anda dapat menentukan semua instans yang menjadi milik dari akun tertentu menggunakan wildcard * seperti berikut ini.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*"
```

Anda juga dapat menentukan semua sumber daya Amazon EC2 yang menjadi milik dari akun tertentu menggunakan wildcard * seperti berikut ini.

```
"Resource": "arn:aws:ec2:us-east-1:123456789012:*"
```

Untuk menentukan semua sumber daya, atau jika tindakan API tertentu tidak mendukung ARN, gunakan wildcard * dalam elemen `Resource` seperti berikut ini.

```
"Resource": "*"
```

Banyak tindakan API Amazon EC2 yang melibatkan beberapa sumber daya. Misalnya, `AttachVolume` melampirkan volume Amazon EBS pada instans, sehingga pengguna harus memiliki izin untuk menggunakan volume dan instans tersebut. Untuk menentukan beberapa sumber daya dalam satu pernyataan, pisahkan ARN sumber daya tersebut menggunakan koma seperti berikut.

```
"Resource": ["arn1", "arn2"]
```

Untuk melihat daftar ARN untuk sumber daya Amazon EC2, lihat Jenis sumber daya yang ditentukan oleh Amazon EC2.

## Kunci syarat untuk Amazon EC2

di pernyataan kebijakan, Anda dapat secara opsional menentukan syarat yang mengontrol kapan pernyataan tersebut berlaku. Setiap syarat mengandung satu atau beberapa pasangan nilai-kunci . Kunci syarat tidak memedulikan huruf besar atau kecil. Kami telah mendefinisikan kunci kondisi AWS global, ditambah kunci kondisi khusus layanan tambahan.

Untuk melihat daftar kunci syarat spesifik layanan untuk Amazon EC2, lihat Kunci syarat untuk Amazon EC2. Amazon EC2 juga mengimplementasikan kunci kondisi AWS global. Untuk informasi selengkapnya, lihat Informasi yang tersedia dalam semua permintaan dalam Panduan Pengguna IAM.

Semua tindakan Amazon EC2 mendukung kunci syarat `aws:RequestedRegion` dan `ec2:Region`. Untuk informasi selengkapnya, lihat Contoh: Membatasi akses ke suatu Wilayah tertentu.

Untuk menggunakan kunci syarat dalam kebijakan IAM Anda, gunakan pernyataan `Condition`. Sebagai contoh, kebijakan berikut memberikan izin kepada para pengguna untuk menambah dan menghapus aturan ke dalam dan ke luar untuk grup keamanan apa pun. Kebijakan tersebut menggunakan kunci syarat `ec2:Vpc` untuk menentukan bahwa tindakan ini hanya dapat dilakukan pada grup keamanan di VPC tertentu.

JSON

```
{
"Version":"2012-10-17",
  "Statement":[{
    "Effect":"Allow",
    "Action": [
       "ec2:AuthorizeSecurityGroupIngress",
       "ec2:AuthorizeSecurityGroupEgress",
       "ec2:RevokeSecurityGroupIngress",
       "ec2:RevokeSecurityGroupEgress"],
     "Resource": "arn:aws:ec2:us-east-1:111122223333:security-group/*",
      "Condition": {
        "StringEquals": {
          "ec2:Vpc": "arn:aws:ec2:us-east-1:111122223333:vpc/
vpc-11223344556677889"
        }
      }
    }
  ]
}
```

Jika Anda menentukan beberapa syarat, atau beberapa kunci dalam satu syarat, maka kami akan mengevaluasinya menggunakan operasi logika AND. Jika Anda menentukan satu syarat dengan beberapa nilai untuk satu kunci, kami akan mengevaluasi syarat tersebut menggunakan operasi logika OR. Agar izin bisa diberikan, semua syarat harus terpenuhi.

Anda juga dapat menggunakan placeholder saat menentukan syarat. Untuk informasi lebih lanjut, lihat Elemen kebijakan IAM: Variabel dan tanda dalam Panduan Pengguna IAM.

**Important**

Beberapa kunci syarat bersifat khusus untuk sumber daya, dan beberapa tindakan API menggunakan beberapa sumber daya. Jika Anda menyusun kebijakan dengan kunci syarat, gunakan elemen `Resource` dari pernyataan tersebut untuk menentukan sumber daya yang padanya kunci syarat tersebut berlaku. Jika tidak, kebijakan ini dapat membuat pengguna tidak bisa melakukan tindakan sama sekali, karena pemeriksaan syarat gagal sebab kunci syarat tidak berlaku terhadap sumber daya tersebut. Jika Anda tidak ingin menentukan sumber daya, atau jika Anda telah menyusun elemen `Action` dari kebijakan

Anda untuk menyertakan beberapa tindakan API, maka Anda harus menggunakan jenis syarat `...IfExists` untuk memastikan bahwa kunci syarat diabaikan untuk sumber daya yang tidak menggunakannya. Untuk informasi lebih lanjut, lihat... IfExists Ketentuan dalam Panduan Pengguna IAM.

Kunci syarat

- EC2: atribut kunci kondisi
- EC2: ResourceId kunci kondisi
- ec2: ARN SourceInstance kunci kondisi

EC2: atribut kunci kondisi

Kunci syarat `ec2:Attribute` dapat digunakan untuk syarat-syarat yang memfilter akses berdasarkan atribut sumber daya.

Kunci kondisi ini hanya mendukung properti yang merupakan tipe data primitif (seperti string atau bilangan bulat), atau AttributeValue objek kompleks yang hanya berisi properti Nilai (seperti Deskripsi atau ImdsSupport objek tindakan ModifyImageAttribute API). Kunci kondisi tidak dapat digunakan dengan objek kompleks yang berisi beberapa properti, seperti LaunchPermission objek dari ModifyImageAttribute.

Misalnya, kebijakan berikut menggunakan kunci `ec2:Attribute/Description` kondisi untuk memfilter akses berdasarkan objek Deskripsi kompleks dari tindakan ModifyImageAttribute API. Kunci syarat hanya mengizinkan permintaan yang memodifikasi deskripsi citra ke `Production` atau `Development`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:ModifyImageAttribute",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
        "StringEquals": {
```

```
          "ec2:Attribute/Description": [
            "Production",
            "Development"
          ]
        }
      }
    }
  ]
}
```

Kebijakan contoh berikut menggunakan kunci `ec2:Attribute` kondisi untuk memfilter akses berdasarkan properti A tribut primitif dari tindakan ModifyImageAttribute API. Kunci syarat menolak semua permintaan yang berusaha memodifikasi deskripsi citra.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Deny",
      "Action": "ec2:ModifyImageAttribute",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
        "StringEquals": {
          "ec2:Attribute": "Description"
        }
      }
    }
  ]
}
```

EC2: ResourceId kunci kondisi

Saat menggunakan kunci syarat `ec2:ResourceID` berikut dengan tindakan API tertentu, nilai kunci syarat digunakan untuk menentukan sumber daya yang dihasilkan yang dibuat oleh tindakan API. Kunci syarat `ec2:ResourceID` tidak dapat digunakan untuk menentukan sumber daya sumber yang ditentukan dalam permintaan API. Jika Anda menggunakan salah satu dari kunci syarat `ec2:ResourceID` berikut dengan API tertentu, maka Anda harus selalu menentukan wildcard (*).

Jika Anda menentukan nilai yang berbeda, syarat tersebut selalu diselesaikan dengan * selama runtime. Misalnya, untuk menggunakan kunci `ec2:ImageId` kondisi dengan CopyImage API, maka Anda harus menentukan kunci kondisi sebagai berikut:

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CopyImage",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-*",
      "Condition": {
        "StringEquals": {
          "ec2:ImageID": "*"
        }
      }
    }
  ]
}
```

Sebaiknya hindari menggunakan kunci kondisi ini dengan tindakan API berikut:

- `ec2:DhcpOptionsID` – `CreateDhcpOptions`
- `ec2:ImageID`—`CopyImage`,`CreateImage`,`ImportImage`, dan `RegisterImage`
- `ec2:InstanceID`— `RunInstances` dan `ImportInstance`
- `ec2:InternetGatewayID` – `CreateInternetGateway`
- `ec2:NetworkAclID` – `CreateNetworkAcl`
- `ec2:NetworkInterfaceID` – `CreateNetworkInterface`
- `ec2:PlacementGroupName` – `CreatePlacementGroup`
- `ec2:RouteTableID` – `CreateRouteTable`
- `ec2:SecurityGroupID` – `CreateSecurityGroup`
- `ec2:SnapshotID`—`CopySnapshot`,`CreateSnapshot`,`CreateSnapshots`, dan `ImportSnapshots`
- `ec2:SubnetID` – `CreateSubnet`

- `ec2:VolumeID`— `CreateVolume` dan `ImportVolume`
- `ec2:VpcID` – `CreateVpc`
- `ec2:VpcPeeringConnectionID` – `CreateVpcPeeringConnection`

Untuk memfilter akses berdasarkan ID sumber daya tertentu, sebaiknya gunakan elemen `Resource` kebijakan sebagai berikut.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:CopyImage",
      "Resource": "arn:aws:ec2:us-east-1::image/ami-01234567890abcdef"
    }
  ]
}
```

ec2: ARN SourceInstance kunci kondisi

Gunakan `ec2:SourceInstanceARN` untuk menentukan ARN dari instance dari mana permintaan dibuat. Ini adalah kunci kondisi AWS global, yang berarti Anda dapat menggunakannya dengan layanan selain Amazon EC2. Untuk contoh kebijakan, lihatContoh: Mengizinkan instance tertentu untuk melihat sumber daya di lainnya AWS layanan.

## Kontrol akses menggunakan akses berbasis atribut

Saat membuat kebijakan IAM yang memberikan izin kepada pengguna untuk menggunakan sumber daya EC2, Anda dapat menyertakan informasi tanda dalam elemen `Condition` dari kebijakan tersebut untuk mengontrol akses berdasarkan tanda. Hal ini dikenal sebagai kendali akses berbasis atribut (ABAC). ABAC memberikan Anda kendali yang lebih baik atas sumber daya mengenai sumber daya mana yang dapat diubah, digunakan, atau dihapus oleh seorang pengguna. Untuk informasi lebih lanjut, lihat Apa fungsi ABAC untuk AWS?

Sebagai contoh, Anda dapat membuat kebijakan yang memungkinkan para pengguna untuk mengakhiri instans, tetapi menolak tindakan itu jika instans tersebut memiliki tanda

`environment=production`. Untuk melakukan hal ini, Anda bisa menggunakan kunci syarat `aws:ResourceTag` untuk mengizinkan atau menolak akses ke sumber daya berdasarkan tanda yang dilampirkan pada sumber daya.

```
"StringEquals": { "aws:ResourceTag/environment": "production" }
```

Untuk mempelajari apakah tindakan API Amazon EC2 mendukung kontrol akses menggunakan kunci syarat `aws:ResourceTag`, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2. Perhatikan bahwa tindakan `Describe` tidak mendukung izin tingkat sumber daya, sehingga Anda harus menentukannya dalam pernyataan terpisah yang tidak disertai syarat.

Untuk contoh kebijakan IAM, lihat Contoh kebijakan untuk mengontrol akses Amazon EC2 API.

Jika Anda mengizinkan atau menolak akses para pengguna ke sumber daya berdasarkan tanda, maka Anda harus mempertimbangkan untuk menolak secara eksplisit memberikan kemampuan kepada pengguna untuk menambahkan atau menghapus tanda tersebut dari sumber daya yang sama. Jika tidak, pengguna dapat mengakali pembatasan Anda dan mendapatkan akses atas sumber daya dengan melakukan modifikasi pada tanda dari sumber daya tersebut.

## Berikan izin kepada pengguna, grup, dan peran

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

  Buat rangkaian izin. Ikuti instruksi di Buat rangkaian izin dalam Panduan Pengguna AWS IAM Identity Center .
- Pengguna yang dikelola di IAM melalui penyedia identitas:

  Buat peran untuk federasi identitas. Ikuti instruksi dalam Buat peran untuk penyedia identitas pihak ketiga (federasi) dalam Panduan Pengguna IAM.
- Pengguna IAM:
  - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam Buat peran untuk pengguna IAM dalam Panduan Pengguna IAM.
  - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam Menambahkan izin ke pengguna (konsol) dalam Panduan Pengguna IAM.

## Memeriksa apakah pengguna memiliki izin yang diperlukan

Setelah Anda membuat kebijakan IAM, kami merekomendasikan agar Anda memeriksa apakah kebijakan itu memberikan izin kepada para pengguna untuk menggunakan tindakan API dan sumber daya tertentu yang mereka butuhkan sebelum Anda memasukkan kebijakan tersebut ke dalam produksi.

Pertama-tama, buatlah pengguna untuk tujuan pengujian, lalu lampirkan kebijakan IAM yang Anda buat tersebut ke pengguna pengujian. Kemudian, buatlah permintaan sebagai pengguna uji.

Jika tindakan Amazon EC2 yang Anda uji membuat atau memodifikasi sumber daya, maka Anda harus mengajukan permintaan menggunakan parameter `DryRun` (atau jalankan perintah AWS CLI dengan opsi `--dry-run`). Dalam hal ini, perintah akan menyelesaikan pemeriksaan otorisasi, tetapi tidak akan menyelesaikan operasi. Sebagai contoh, Anda dapat memeriksa apakah pengguna dapat mengakhiri instans tertentu tanpa benar-benar mengakhirinya. Jika pengguna uji tersebut memiliki izin yang diperlukan, maka permintaan itu akan menampilkan `DryRunOperation`; jika tidak, `UnauthorizedOperation` yang akan ditampilkan.

Jika kebijakan tersebut tidak memberikan izin kepada pengguna seperti yang Anda harapkan, atau terlalu longgar dalam memberikan izin, maka Anda dapat menyesuaikan kebijakan itu sesuai kebutuhan Anda dan menguji ulang hingga Anda mendapatkan hasil yang Anda inginkan.

> **Important**
>
> Pengujian ini dapat memakan waktu beberapa menit sebelum perubahan terjadi pada kebijakan untuk ditransmisikan sebelum diberlakukan. Oleh karena itu, kami merekomendasikan Anda memberikan waktu lima menit sebelum Anda menguji pembaruan kebijakan Anda.

Jika pemeriksaan otorisasi gagal, maka permintaan akan menampilkan informasi berenkode yang memuat informasi diagnostik. Anda dapat melakukan dekode pada pesan tersebut menggunakan tindakan `DecodeAuthorizationMessage`. Untuk informasi selengkapnya, lihat DecodeAuthorizationMessage di Referensi AWS Security Token Service API, dan deco de-authorization-message.

# Contoh kebijakan untuk mengontrol akses Amazon EC2 API

Anda dapat menggunakan kebijakan IAM untuk memberi pengguna izin yang diperlukan untuk bekerja dengan Amazon EC2. Untuk petunjuk langkah demi langkah, lihat Membuat kebijakan IAM di Panduan Pengguna IAM.

Contoh berikut menunjukkan pernyataan kebijakan yang dapat Anda gunakan untuk memberikan izin kepada pengguna untuk menggunakan Amazon EC2. Kebijakan ini dirancang untuk permintaan yang dibuat menggunakan AWS CLI atau AWS SDK. Dalam contoh berikut, ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri.

Contoh

- Contoh: Read-only akses
- Contoh: Membatasi akses ke suatu Wilayah tertentu
- Cara menggunakan instans
- Luncurkan instance (RunInstances)
- Cara Menggunakan Instans Spot
- Contoh: Cara Menggunakan Instans Cadangan
- Contoh: Memberi tanda pada sumber daya
- Contoh: Cara Menggunakan IAM role
- Contoh: Cara menggunakan tabel rute
- Contoh: Mengizinkan instance tertentu untuk melihat sumber daya di lainnya AWS layanan
- Contoh: Cara menggunakan templat peluncuran
- Cara menggunakan metadata instans
- Bekerja dengan volume dan snapshot Amazon EBS

Untuk contoh kebijakan yang digunakan dalam konsol Amazon EC2, lihat Contoh kebijakan untuk mengontrol akses ke konsol Amazon EC2.

## Contoh: Read-only akses

Kebijakan berikut memberikan izin kepada para pengguna untuk menggunakan semua tindakan API Amazon EC2 yang memiliki nama dimulai dengan `Describe`. Elemen `Resource` menggunakan wildcard untuk mengindikasikan bahwa para pengguna dapat menentukan semua sumber daya dengan tindakan API ini. Wildcard * juga Anda perlukan jika tindakan API tidak mendukung izin

tingkat sumber daya. Untuk informasi selengkapnya tentang ARN mana yang dapat Anda gunakan bersama dengan tindakan API Amazon EC2 yang mana, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2.

Para pengguna tidak memiliki izin untuk melakukan tindakan apa pun atas sumber daya (kecuali jika pernyataan lain memberikan mereka izin untuk melakukannya) karena mereka tidak mendapatkan izin untuk menggunakan tindakan API secara default.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:Describe*",
      "Resource": "*"
    }
   ]
}
```

## Contoh: Membatasi akses ke suatu Wilayah tertentu

Kebijakan berikut menolak pemberian izin kepada para pengguna untuk menggunakan semua tindakan API Amazon EC2 kecuali Wilayah tersebut adalah Eropa (Frankfurt). Kebijakan ini menggunakan kunci syarat global `aws:RequestedRegion`, yang didukung oleh semua tindakan API Amazon EC2.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
       {
      "Effect": "Deny",
      "Action": "ec2:*",
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
```

```
          "aws:RequestedRegion": "eu-central-1"
        }
      }
    }
  ]
}
```

Atau, Anda dapat menggunakan kunci syarat `ec2:Region`, yang dikhususkan untuk Amazon EC2 dan didukung oleh semua tindakan API Amazon EC2.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
       {
      "Effect": "Deny",
      "Action": "ec2:*",
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "ec2:Region": "eu-central-1"
        }
      }
    }
  ]
}
```

## Cara menggunakan instans

Contoh

- Contoh: Mendeskripsikan, meluncurkan, menghentikan, memulai, dan mengakhiri semua instans
- Contoh: Mendeskripsikan semua instans, dan menghentikan, memulai, dan mengakhiri instans tertentu saja

Contoh: Mendeskripsikan, meluncurkan, menghentikan, memulai, dan mengakhiri semua instans

Kebijakan berikut memberikan izin kepada para pengguna untuk menggunakan tindakan API yang ditentukan dalam elemen `Action`. Elemen `Resource` menggunakan wildcard * untuk mengindikasikan bahwa para pengguna dapat menentukan semua sumber daya dengan tindakan API ini. Wildcard * juga Anda perlukan jika tindakan API tidak mendukung izin tingkat sumber daya. Untuk informasi selengkapnya tentang ARN mana yang dapat Anda gunakan bersama dengan tindakan API Amazon EC2 yang mana, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2.

Para pengguna tidak memiliki izin untuk menggunakan tindakan API apa pun (kecuali jika pernyataan lain memberikan mereka izin untuk melakukannya) karena para pengguna tersebut tidak mendapatkan izin untuk menggunakan tindakan API secara default.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeInstances",
        "ec2:DescribeImages",
        "ec2:DescribeKeyPairs",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeAvailabilityZones",
        "ec2:RunInstances",
        "ec2:TerminateInstances",
        "ec2:StopInstances",
        "ec2:StartInstances"
      ],
      "Resource": "*"
    }
   ]
}
```

Contoh: Mendeskripsikan semua instans, dan menghentikan, memulai, dan mengakhiri instans tertentu saja

Kebijakan berikut memungkinkan pengguna untuk menjelaskan semua instans, untuk memulai dan menghentikan hanya instans i-1234567890abcdef0 dan i-0598c7d356eba48d7, dan untuk menghentikan hanya instance di Wilayah, dengan tag sumber daya "". `us-east-1` `purpose=test`

Pernyataan pertama menggunakan wildcard * untuk elemen `Resource` untuk mengindikasikan bahwa para pengguna dapat menentukan semua sumber daya dengan tindakan tersebut; dalam hal ini, mereka dapat mencantumkan semua instans. Wildcard * tersebut juga Anda perlukan jika tindakan API tidak mendukung izin tingkat sumber daya (dalam hal ini, `ec2:DescribeInstances`). Untuk informasi selengkapnya tentang ARN mana yang dapat Anda gunakan bersama dengan tindakan API Amazon EC2 yang mana, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2.

Pernyataan kedua menggunakan izin tingkat sumber daya untuk tindakan `StopInstances` dan `StartInstances`. Instans-instans tertentu diindikasikan oleh ARN yang mereka punya dalam elemen `Resource`.

Pernyataan ketiga memungkinkan pengguna untuk menghentikan semua instance di `us-east-1` Wilayah yang termasuk dalam AWS akun yang ditentukan, tetapi hanya jika instance memiliki tag`"purpose=test"`. Elemen `Condition` memenuhi syarat ketika pernyataan kebijakan berlaku.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
   {
   "Effect": "Allow",
      "Action": "ec2:DescribeInstances",
      "Resource": "*"
   },
   {
      "Effect": "Allow",
      "Action": [
        "ec2:StopInstances",
        "ec2:StartInstances"
      ],
      "Resource": [
```

```
        "arn:aws:ec2:us-east-1:111122223333:instance/i-1234567890abcdef0",
        "arn:aws:ec2:us-east-1:111122223333:instance/i-0598c7d356eba48d7"
      ]
    },
    {
      "Effect": "Allow",
      "Action": "ec2:TerminateInstances",
      "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
      "Condition": {
         "StringEquals": {
            "aws:ResourceTag/purpose": "test"
         }
      }
   }

   ]
}
```

## Luncurkan instance (RunInstances)

Tind RunInstances akan API meluncurkan satu atau beberapa On-Demand Instans atau satu atau beberapa Instans Spot. `RunInstances`membutuhkan AMI dan membuat instance. Para pengguna dapat menentukan pasangan kunci dan grup keamanan dalam permintaan. Peluncuran ke dalam VPC memerlukan subnet, dan akan membuat antarmuka jaringan. Meluncurkan dari EBS-backed AMI Amazon membuat volume. Oleh karena itu, para pengguna harus memiliki izin untuk menggunakan sumber daya Amazon EC2 ini. Anda dapat membuat pernyataan kebijakan yang mengharuskan pengguna menentukan parameter opsional pada `RunInstances`, atau membatasi pengguna pada nilai tertentu sebagai parameter.

Untuk informasi selengkapnya tentang izin tingkat sumber daya yang diperlukan untuk meluncurkan instans, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2.

Secara default, para pengguna tidak memiliki izin untuk mendeskripsikan, memulai, menghentikan, atau mengakhiri instans yang dihasilkan. Salah satu cara untuk memberikan izin kepada para pengguna untuk mengelola instans yang dihasilkan adalah dengan membuat tanda tertentu untuk setiap instans, dan kemudian membuat pernyataan yang memungkinkan mereka mengelola instans-instans itu dengan tanda tersebut. Untuk informasi selengkapnya, lihat Cara menggunakan instans.

Sumber daya

- AMI

- Tipe instans
- Subnet
- Volume EBS
- Tanda
- Tanda di templat peluncuran
- GPU elastis
- Templat peluncuran

AMI

Kebijakan berikut memungkinkan para pengguna untuk meluncurkan instans hanya menggunakan AMI yang ditentukan, `ami-9e1670f7` dan `ami-45cf5c3c`. Pengguna tidak dapat meluncurkan instans dengan AMI lain (kecuali jika pernyataan lain memberi pengguna izin untuk melakukannya).

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:us-east-1::image/ami-9e1670f7",
        "arn:aws:ec2:us-east-1::image/ami-45cf5c3c",
        "arn:aws:ec2:us-east-1:111122223333:instance/*",
        "arn:aws:ec2:us-east-1:111122223333:volume/*",
        "arn:aws:ec2:us-east-1:111122223333:key-pair/*",
        "arn:aws:ec2:us-east-1:111122223333:security-group/*",
        "arn:aws:ec2:us-east-1:111122223333:subnet/*",
        "arn:aws:ec2:us-east-1:111122223333:network-interface/*"
      ]
    }
   ]
}
```

Atau, kebijakan berikut memungkinkan pengguna untuk meluncurkan instans dari semua AMI yang dimiliki oleh Amazon, atau partner lain yang tepercaya dan terverifikasi. Elemen `Condition`

dari pernyataan pertama menguji apakah `ec2:Owner` adalah `amazon`. Pengguna tidak dapat meluncurkan instans dengan AMI lain (kecuali jika pernyataan lain memberi pengguna izin untuk melakukannya).

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
         {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1::image/ami-*"
      ],
      "Condition": {
         "StringEquals": {
            "ec2:Owner": "amazon"
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1:111122223333:instance/*",
         "arn:aws:ec2:us-east-1:111122223333:subnet/*",
         "arn:aws:ec2:us-east-1:111122223333:volume/*",
         "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
         "arn:aws:ec2:us-east-1:111122223333:key-pair/*",
         "arn:aws:ec2:us-east-1:111122223333:security-group/*"
         ]
      }
   ]
}
```

Tipe instans

Kebijakan berikut memungkinkan para pengguna untuk meluncurkan instans dengan hanya menggunakan tipe instans `t2.micro` atau `t2.small`, hal ini mungkin Anda lakukan untuk mengontrol biaya. Para pengguna tidak dapat meluncurkan instans yang lebih besar karena elemen

`Condition` dari pernyataan pertama menguji apakah `ec2:InstanceType` merupakan `t2.micro` atau `t2.small`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
        {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1:111122223333:instance/*"
      ],
      "Condition": {
         "StringEquals": {
            "ec2:InstanceType": ["t2.micro", "t2.small"]
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1::image/ami-*",
         "arn:aws:ec2:us-east-1:111122223333:subnet/*",
         "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
         "arn:aws:ec2:us-east-1:111122223333:volume/*",
         "arn:aws:ec2:us-east-1:111122223333:key-pair/*",
         "arn:aws:ec2:us-east-1:111122223333:security-group/*"
         ]
      }
   ]
}
```

Atau, Anda dapat membuat kebijakan yang menolak memberikan izin kepada pengguna untuk meluncurkan instans apa pun kecuali tipe instans `t2.micro` dan `t2.small`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
        {
      "Effect": "Deny",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1:111122223333:instance/*"
      ],
      "Condition": {
         "StringNotEquals": {
            "ec2:InstanceType": ["t2.micro", "t2.small"]
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1::image/ami-*",
         "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
         "arn:aws:ec2:us-east-1:111122223333:instance/*",
         "arn:aws:ec2:us-east-1:111122223333:subnet/*",
         "arn:aws:ec2:us-east-1:111122223333:volume/*",
         "arn:aws:ec2:us-east-1:111122223333:key-pair/*",
         "arn:aws:ec2:us-east-1:111122223333:security-group/*"
         ]
      }
   ]
}
```

Subnet

Kebijakan berikut memungkinkan para pengguna untuk meluncurkan instans hanya menggunakan subnet yang ditentukan, `subnet-12345678`. Grup tidak dapat meluncurkan instans ke subnet lain mana pun (kecuali pernyataan lain memberikan izin kepada pengguna untuk melakukannya).

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
        "arn:aws:ec2:us-east-1:111122223333:subnet/subnet-12345678",
        "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
        "arn:aws:ec2:us-east-1:111122223333:instance/*",
        "arn:aws:ec2:us-east-1:111122223333:volume/*",
        "arn:aws:ec2:us-east-1::image/ami-*",
        "arn:aws:ec2:us-east-1:111122223333:key-pair/*",
        "arn:aws:ec2:us-east-1:111122223333:security-group/*"
      ]
    }
   ]
}
```

Atau, Anda dapat membuat kebijakan yang menolak memberikan izin kepada pengguna untuk meluncurkan instans ke dalam subnet lain mana pun. Pernyataan tersebut menjalankan hal ini dengan menolak memberikan izin untuk membuat antarmuka jaringan, kecuali jika subnet `subnet-12345678` telah ditentukan. Penolakan ini akan mengabaikan kebijakan lain yang dibuat untuk memungkinkan peluncuran instans ke dalam subnet lain.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
         {
      "Effect": "Deny",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1:111122223333:network-interface/*"
      ],
      "Condition": {
         "ArnNotEquals": {
```

```
            "ec2:Subnet": "arn:aws:ec2:us-east-1:111122223333:subnet/
subnet-12345678"
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-1::image/ami-*",
         "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
         "arn:aws:ec2:us-east-1:111122223333:instance/*",
         "arn:aws:ec2:us-east-1:111122223333:subnet/*",
         "arn:aws:ec2:us-east-1:111122223333:volume/*",
         "arn:aws:ec2:us-east-1:111122223333:key-pair/*",
         "arn:aws:ec2:us-east-1:111122223333:security-group/*"
         ]
      }
   ]
}
```

Volume EBS

Kebijakan berikut memungkinkan para pengguna untuk meluncurkan instans hanya jika volume EBS untuk instans tersebut sudah dienkripsi. Pengguna harus meluncurkan instans dari AMI yang dibuat dengan snapshot yang dienkripsi untuk memastikan volume root sudah dienkripsi. Volume tambahan apa pun yang dilampirkan oleh pengguna pada instans saat dilakukan peluncuran juga harus sudah dienkripsi.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
                {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:*:*:volume/*"
            ],
            "Condition": {
```

```
                "Bool": {
                    "ec2:Encrypted": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:*::image/ami-*",
                "arn:aws:ec2:*:*:network-interface/*",
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ec2:*:*:subnet/*",
                "arn:aws:ec2:*:*:key-pair/*",
                "arn:aws:ec2:*:*:security-group/*"
            ]
        }
    ]
}
```

Tanda

Memberi tanda pada instans pada saat instans dibuat

Kebijakan berikut memungkinkan para pengguna untuk meluncurkan instans dan memberi tanda pada instans saat instans sedang dibuat. Untuk tindakan-tindakan yang digunakan untuk membuat sumber daya yang menerapkan tanda, para pengguna harus memiliki izin untuk menggunakan tindakan `CreateTags`. Pernyataan kedua menggunakan kunci syarat `ec2:CreateAction` untuk memungkinkan para pengguna membuat tanda hanya dalam konteks `RunInstances`, dan hanya untuk instans. Para pengguna tidak dapat memberi tanda pada sumber daya yang sudah ada, dan para pengguna tidak dapat memberi tanda pada volume menggunakan permintaan `RunInstances`.

Untuk informasi selengkapnya, lihat Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
```

```
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
      "Condition": {
         "StringEquals": {
             "ec2:CreateAction" : "RunInstances"
          }
       }
    }
  ]
}
```

Memberi tanda pada instans dan volume pada saat pembuatan dengan tanda tertentu

Kebijakan berikut mencakup kunci syarat `aws:RequestTag` yang mengharuskan para pengguna untuk menandai setiap instans dan volume yang dibuat oleh `RunInstances` dengan tanda `environment=production` dan `purpose=webserver`. Jika pengguna tidak meneruskan tanda tertentu ini, atau jika mereka tidak menentukan tanda sama sekali, maka permintaan itu akan gagal.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
   {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": [
         "arn:aws:ec2:us-east-1::image/*",
         "arn:aws:ec2:us-east-1:111122223333:subnet/*",
```

```
            "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
            "arn:aws:ec2:us-east-1:111122223333:security-group/*",
            "arn:aws:ec2:us-east-1:111122223333:key-pair/*"
        ]
    },
    {
        "Effect": "Allow",
        "Action": [
            "ec2:RunInstances"
        ],
        "Resource": [
            "arn:aws:ec2:us-east-1:111122223333:volume/*",
            "arn:aws:ec2:us-east-1:111122223333:instance/*"
        ],
        "Condition": {
            "StringEquals": {
                "aws:RequestTag/environment": "production" ,
                "aws:RequestTag/purpose": "webserver"
            }
        }
    },
    {
        "Effect": "Allow",
        "Action": [
            "ec2:CreateTags"
        ],
        "Resource": "arn:aws:ec2:us-east-1:111122223333:*/*",
        "Condition": {
            "StringEquals": {
                "ec2:CreateAction" : "RunInstances"
            }
        }
    }
  ]
}
```

Memberi tanda pada instans dan volume pada saat pembuatan dengan setidaknya satu tanda tertentu

Kebijakan berikut menggunakan pemodifikasi `ForAnyValue` berdasarkan syarat `aws:TagKeys` untuk mengindikasikan bahwa setidaknya satu tanda harus ditentukan dalam permintaan, dan harus

berisi kunci `environment` atau `webserver`. Tanda harus diterapkan baik untuk instans maupun volume. Nilai tanda apa pun juga dapat ditentukan dalam permintaan.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
   {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": [
         "arn:aws:ec2:us-east-1::image/*",
         "arn:aws:ec2:us-east-1:111122223333:subnet/*",
         "arn:aws:ec2:us-east-1:111122223333:network-interface/*",
         "arn:aws:ec2:us-east-1:111122223333:security-group/*",
         "arn:aws:ec2:us-east-1:111122223333:key-pair/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
          "ec2:RunInstances"
      ],
      "Resource": [
          "arn:aws:ec2:us-east-1:111122223333:volume/*",
          "arn:aws:ec2:us-east-1:111122223333:instance/*"
      ],
      "Condition": {
          "ForAnyValue:StringEquals": {
              "aws:TagKeys": ["environment","webserver"]
          }
       }
    },
    {
      "Effect": "Allow",
      "Action": [
          "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:*/*",
      "Condition": {
```

```
            "StringEquals": {
                "ec2:CreateAction" : "RunInstances"
            }
        }
    }
  ]
}
```

Jika instans diberi tanda pada saat instans dibuat, maka instans tersebut harus diberi tanda dengan tanda tertentu

Dalam kebijakan berikut, para pengguna tidak perlu menentukan tanda dalam permintaan, tetapi jika mereka melakukannya, tanda harus berupa `purpose=test`. Tidak ada tanda lain yang diperbolehkan. Pengguna dapat menerapkan tanda ke sumber daya mana pun yang dapat diberi tanda dalam permintaan `RunInstances`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:*/*",
      "Condition": {
         "StringEquals": {
             "aws:RequestTag/purpose": "test",
             "ec2:CreateAction" : "RunInstances"
          },
          "ForAllValues:StringEquals": {
              "aws:TagKeys": "purpose"
```

```
          }
       }
    }
  ]
}
```

Untuk melarang siapa pun yang memanggil tag pada create for RunInstances

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Effect": "Deny",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

Hanya izinkan tanda tertentu untuk spot-instances-request. Inkonsistensi kejutan nomor 2 akan mempengaruhi hasilnya. Dalam keadaan normal, tidak menentukan tanda akan menghasilkan Tidak terautentikasi. Dalam hal spot-instances-request, kebijakan ini tidak akan dievaluasi jika tidak ada tanda spot-instances-request, jadi permintaan Spot on Run tanpa tanda akan berhasil.

Tanda di templat peluncuran

Dalam contoh berikut ini, para pengguna dapat meluncurkan beberapa instans, tetapi hanya jika mereka menggunakan templat peluncuran tertentu (`lt-09477bcd97b0d310e`). Kunci syarat `ec2:IsLaunchTemplateResource` mencegah para pengguna untuk mengganti sumber daya apa pun yang ditentukan dalam templat peluncuran tersebut. Bagian kedua dari pernyataan ini memungkinkan para pengguna untuk memberikan tanda pada instans saat instans dibuat—bagian pernyataan ini diperlukan jika tanda ditentukan untuk instans dalam templat peluncuran.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:us-east-1:111122223333:launch-
template/lt-09477bcd97b0d310e"
          },
          "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
      "Condition": {
         "StringEquals": {
```

```
            "ec2:CreateAction" : "RunInstances"
          }
        }
      }
    ]
}
```

GPU elastis

Dalam kebijakan berikut, para pengguna dapat meluncurkan instans dan menentukan GPU elastis untuk dilampirkan ke instans. Pengguna dapat meluncurkan instans di Wilayah mana pun, tetapi mereka hanya dapat melampirkan GPU elastis saat dilakukan peluncuran dalam Wilayah `us-east-2`.

Kunci syarat `ec2:ElasticGpuType` memastikan bahwa instans menggunakan tipe GPU elastis `eg1.medium` atau `eg1.large`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
             {
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:*:111122223333:elastic-gpu/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:Region": "us-east-2",
                    "ec2:ElasticGpuType": [
                        "eg1.medium",
                        "eg1.large"
                    ]
                }
            }
        },
        {
```

```
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": [
                "arn:aws:ec2:*::image/ami-*",
                "arn:aws:ec2:*:111122223333:network-interface/*",
                "arn:aws:ec2:*:111122223333:instance/*",
                "arn:aws:ec2:*:111122223333:subnet/*",
                "arn:aws:ec2:*:111122223333:volume/*",
                "arn:aws:ec2:*:111122223333:key-pair/*",
                "arn:aws:ec2:*:111122223333:security-group/*"
            ]
        }
    ]
}
```

Templat peluncuran

Dalam contoh berikut ini, para pengguna dapat meluncurkan beberapa instans, tetapi hanya jika mereka menggunakan templat peluncuran tertentu (`lt-09477bcd97b0d310e`). Para pengguna dapat mengganti parameter apa pun dalam templat peluncuran itu dengan menentukan parameter dalam tindakan `RunInstances`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
        {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:us-east-1:111122223333:launch-
template/lt-09477bcd97b0d310e"
          }
       }
    }
  ]
}
```

Dalam contoh berikut, para pengguna dapat meluncurkan instans hanya jika mereka menggunakan templat peluncuran. Kebijakan ini menggunakan kunci syarat `ec2:IsLaunchTemplateResource` untuk mencegah para pengguna mengganti ARN yang sudah ada sebelumnya dalam templat peluncuran.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
         {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:us-east-1:111122223333:launch-
template/*"
          },
          "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    }
  ]
}
```

Contoh kebijakan berikut ini memungkinkan para pengguna untuk meluncurkan instans, tetapi hanya jika mereka menggunakan templat peluncuran. Para pengguna tidak dapat mengganti parameter subnet dan antarmuka jaringan dalam permintaan; parameter-parameter ini hanya dapat ditentukan dalam templat peluncuran. Bagian pertama pernyataan menggunakan elemen NotResource untuk memungkinkan semua sumber daya kecuali subnet dan antarmuka jaringan. Bagian kedua dari pernyataan mengizinkan sumber daya subnet dan antarmuka jaringan, tetapi hanya jika sumber tersebut berasal dari templat peluncuran.

JSON

```
{
  "Version":"2012-10-17",
```

```
  "Statement": [
        {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "NotResource": ["arn:aws:ec2:us-east-1:111122223333:subnet/*",
                      "arn:aws:ec2:us-east-1:111122223333:network-interface/*" ],
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:us-east-1:111122223333:launch-
template/*"
          }
       }
    },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": ["arn:aws:ec2:us-east-1:111122223333:subnet/*",
                   "arn:aws:ec2:us-east-1:111122223333:network-interface/*" ],
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:us-east-1:111122223333:launch-
template/*"
          },
          "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    }
  ]
}
```

Contoh berikut ini memungkinkan para pengguna untuk meluncurkan instans hanya jika mereka menggunakan templat peluncuran, dan hanya jika templat peluncuran memiliki tanda `Purpose=Webservers`. Para pengguna tidak dapat mengganti parameter templat peluncuran dalam tindakan `RunInstances`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
```

```
        {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "NotResource": "arn:aws:ec2:us-east-1:111122223333:launch-template/*",
      "Condition": {
         "ArnLike": {
             "ec2:LaunchTemplate": "arn:aws:ec2:us-east-1:111122223333:launch-
template/*"
          },
         "Bool": {
             "ec2:IsLaunchTemplateResource": "true"
          }
       }
    },
    {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": "arn:aws:ec2:us-east-1:111122223333:launch-template/*",
      "Condition": {
       "StringEquals": {
           "aws:ResourceTag/Purpose": "Webservers"
        }
       }
     }
  ]
}
```

## Cara Menggunakan Instans Spot

Anda dapat menggunakan RunInstances tindakan untuk membuat permintaan Instans Spot, dan menandai permintaan Instance Spot saat membuat. Sumber daya yang harus ditentukan RunInstances adalah`spot-instances-request`.

Sumber daya `spot-instances-request` dievaluasi dalam kebijakan IAM sebagaimana berikut ini:

- Jika Anda tidak memberi tag pada permintaan Instans Spot saat dibuat, Amazon EC2 tidak mengevaluasi`spot-instances-request` sumber daya di RunInstances pernyataan.
- Jika Anda memberi tag pada permintaan Instans Spot saat dibuat, Amazon EC2 mengevaluasi`spot-instances-request` sumber daya di RunInstances pernyataan.

Oleh karena itu, untuk sumber daya `spot-instances-request`, aturan-aturan berikut berlaku untuk kebijakan IAM:

- Jika Anda menggunakan RunInstances untuk membuat permintaan Instance Spot dan Anda tidak bermaksud untuk menandai permintaan Instans Spot saat dibuat, Anda tidak perlu mengizinkan `spot-instances-request` sumber daya secara eksplisit; panggilan akan berhasil.
- Jika Anda menggunakan RunInstances untuk membuat permintaan Instance Spot dan ingin menandai permintaan Instans Spot saat dibuat, Anda harus menyertakan `spot-instances-request` sumber daya dalam pernyataan RunInstances allow, jika tidak panggilan akan gagal.
- Jika Anda menggunakan RunInstances untuk membuat permintaan Instance Spot dan ingin menandai permintaan Instance Spot saat dibuat, Anda harus menentukan `spot-instances-request` resource atau * wildcard dalam pernyataan CreateTags allow, jika tidak panggilan akan gagal.

Anda dapat meminta Instans Spot menggunakan RunInstances atau RequestSpotInstances. Contoh kebijakan IAM berikut hanya berlaku saat meminta Instans Spot menggunakan RunInstances.

Contoh: Meminta Instans Spot menggunakan RunInstances

Kebijakan berikut memungkinkan pengguna meminta Instans Spot dengan menggunakan RunInstances tindakan tersebut. Sumber `spot-instances-request` daya, yang dibuat oleh RunInstances, meminta Instans Spot.

> **Note**
>
> Untuk digunakan RunInstances untuk membuat permintaan Instans Spot, Anda dapat menghilangkan `spot-instances-request` dari `Resource` daftar jika Anda tidak bermaksud untuk menandai permintaan Instans Spot saat dibuat. Hal ini dikarenakan Amazon EC2 tidak mengevaluasi sumber daya `spot-instances-request` dalam pernyataan RunInstances jika permintaan Instans Spot tidak diberi tag pada saat penciptaan.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
```

```
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        }
    ]
}
```

**Warning**

TIDAK DIDUKUNG - Contoh: Tolak izin pengguna untuk meminta Instans Spot menggunakan RunInstances

Kebijakan berikut ini tidak mendukung sumber daya `spot-instances-request`.

Kebijakan berikut dimaksudkan untuk memberi pengguna izin untuk meluncurkan On-Demand Instans, tetapi menolak pengguna izin untuk meminta Instans Spot. Sumber `spot-instances-request` daya, yang dibuat oleh RunInstances, adalah sumber daya yang meminta Instans Spot. Pernyataan kedua dimaksudkan untuk menolak RunInstances tindakan untuk `spot-instances-request` sumber daya. Namun, kondisi ini tidak didukung karena Amazon EC2 tidak mengevaluasi `spot-instances-request` sumber daya dalam RunInstances pernyataan jika permintaan Instans Spot tidak diberi tag saat dibuat.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
```

```
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*"
            ]
        },
        {
            "Sid": "DenySpotInstancesRequestsNOTSUPPORTEDDONOTUSE",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
        }
    ]
}
```

Contoh: Memberikan tanda pada permintaan Instans Spot pada saat dibuat

Kebijakan berikut ini memungkinkan para pengguna untuk memberikan tanda pada semua sumber daya yang dibuat saat dilakukan peluncuran instans. Pernyataan pertama memungkinkan RunInstances untuk membuat sumber daya yang terdaftar. Sumber `spot-instances-request` daya, yang dibuat oleh RunInstances, adalah sumber daya yang meminta Instans Spot. Pernyataan kedua menyediakan wildcard * untuk mengizinkan semua sumber daya diberi tanda pada saat dibuat ketika peluncuran instans.

> **Note**
>
> Jika Anda menandai permintaan Instans Spot saat dibuat, Amazon EC2 mengevaluasi `spot-instances-request` sumber daya dalam RunInstances pernyataan tersebut. Oleh karena itu, Anda harus secara eksplisit mengi `spot-instances-request` zinkan sumber daya untuk RunInstances tindakan tersebut, jika tidak panggilan akan gagal.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "TagResources",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

Contoh: Menolak tanda pada saat dibuat untuk permintaan Instans Spot

Kebijakan berikut ini menolak memberikan izin kepada para pengguna untuk memberikan tanda pada semua sumber daya yang dibuat saat dilakukan peluncuran instans.

Pernyataan pertama memungkinkan RunInstances untuk membuat sumber daya yang terdaftar. Sumber `spot-instances-request` daya, yang dibuat oleh RunInstances, adalah sumber daya yang meminta Instans Spot. Pernyataan kedua menyediakan wildcard * untuk menolak semua sumber daya yang sedang diberi tanda pada saat dibuat ketika peluncuran instans. Jika

`spot-instances-request` atau sumber daya lain diberi tag pada saat penciptaan, perintah RunInstances akan gagal.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "DenyTagResources",
            "Effect": "Deny",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

**Warning**

TIDAK DIDUKUNG - Contoh: Mengizinkan pembuatan permintaan Instans Spot hanya jika ada tanda khusus yang ditetapkan untuknya
Kebijakan berikut ini tidak mendukung sumber daya `spot-instances-request`.

Kebijakan berikut dimaksudkan untuk memberikan izin RunInstances untuk membuat permintaan Instans Spot hanya jika permintaan diberi tag dengan tag tertentu. Pernyataan pertama memungkinkan RunInstances untuk membuat sumber daya yang terdaftar.
Pernyataan kedua dimaksudkan untuk memberikan izin kepada para pengguna untuk membuat permintaan Instans Spot hanya jika permintaan itu memiliki tanda `environment=production`. Jika kondisi ini diterapkan ke sumber daya lain yang dibuat oleh RunInstances, menentukan tidak ada tag menghasilkan `Unauthenticated` kesalahan. Namun, jika tidak ada tag yang ditentukan untuk permintaan Instans Spot, Amazon EC2 tidak mengevaluasi `spot-instances-request` sumber daya dalam RunInstances pernyataan, yang mengakibatkan permintaan Instans Spot yang tidak diberi tag dibuat oleh. RunInstances Perhatikan bahwa penentuan tag lain selain `environment=production` akan menghasilkan eror `Unauthenticated` karena jika pengguna memberi tag pada permintaan Instans Spot, Amazon EC2 akan mengevaluasi sumber daya`spot-instances-request` dalam pernyataan RunInstances .

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*"
            ]
        },
        {
            "Sid":
 "RequestSpotInstancesOnlyIfTagIsEnvironmentProductionNOTSUPPORTEDDONOTUSE",
```

```
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/
*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "production"
                }
            }
        },
        {
            "Sid": "TagResources",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }

    ]
}
```

Contoh: Menolak membuat permintaan Instans Spot jika ada tanda tertentu yang ditetapkan untuknya

Kebijakan berikut menolak izin RunInstances untuk membuat permintaan Instans Spot jika permintaan diberi tag dengan`environment=production`.

Pernyataan pertama memungkinkan RunInstances untuk membuat sumber daya yang terdaftar.

Pernyataan kedua menolak memberikan izin kepada para pengguna untuk membuat permintaan Instans Spot jika permintaan itu memiliki tanda `environment=production`. Menentukan `environment=production` sebagai tanda akan mengakibatkan munculnya kesalahan `Unauthenticated`. Menentukan tanda lain atau tidak menentukan tanda akan mengakibatkan terciptanya permintaan Instans Spot.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowRun",
```

```
            "Effect": "Allow",
            "Action": [
                "ec2:RunInstances"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1::image/*",
                "arn:aws:ec2:us-east-1:*:subnet/*",
                "arn:aws:ec2:us-east-1:*:network-interface/*",
                "arn:aws:ec2:us-east-1:*:security-group/*",
                "arn:aws:ec2:us-east-1:*:key-pair/*",
                "arn:aws:ec2:us-east-1:*:volume/*",
                "arn:aws:ec2:us-east-1:*:instance/*",
                "arn:aws:ec2:us-east-1:*:spot-instances-request/*"
            ]
        },
        {
            "Sid": "DenySpotInstancesRequests",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:us-east-1:*:spot-instances-request/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "production"
                }
            }
        },
        {
            "Sid": "TagResources",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*"
        }
    ]
}
```

## Contoh: Cara Menggunakan Instans Cadangan

Kebijakan berikut ini memberikan izin kepada para pengguna untuk menampilkan, memodifikasi, dan membeli Instans Cadangan dalam akun Anda.

Pengaturan izin tingkat sumber daya untuk masing-masing Instans Cadangan tidak bisa dilakukan. Kebijakan ini berarti para pengguna memiliki akses ke semua Instans Cadangan dalam akun tersebut.

Elemen `Resource` menggunakan wildcard * untuk mengindikasikan bahwa para pengguna dapat menentukan semua sumber daya dengan tindakan; dalam hal ini, mereka dapat mencantumkan dan memodifikasi semua Instans Cadangan dalam akun. Mereka juga dapat membeli Instans Cadangan menggunakan kredensial akun. Wildcard * juga Anda perlukan jika tindakan API tidak mendukung izin tingkat sumber daya.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeReservedInstances",
        "ec2:ModifyReservedInstances",
        "ec2:PurchaseReservedInstancesOffering",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeReservedInstancesOfferings"
      ],
      "Resource": "*"
    }
   ]
}
```

Untuk memungkinkan para pengguna menampilkan dan memodifikasi Instans Cadangan dalam akun Anda, tetapi tidak untuk membeli Instans Cadangan baru.

JSON

```
{
  "Version":"2012-10-17",
   "Statement": [
    {
      "Effect": "Allow",
```

```
      "Action": [
        "ec2:DescribeReservedInstances",
        "ec2:ModifyReservedInstances",
        "ec2:DescribeAvailabilityZones"
      ],
      "Resource": "*"
    }
  ]
}
```

## Contoh: Memberi tanda pada sumber daya

Kebijakan berikut ini memungkinkan para pengguna untuk menggunakan tindakan `CreateTags` untuk menerapkan tanda ke instans hanya jika tanda tersebut berisi kunci `environment` dan nilai `production`. Tidak ada tag lain yang diizinkan dan pengguna tidak dapat menandai jenis sumber daya lainnya.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
              {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/environment": "production"
                }
            }
        }
    ]
}
```

Kebijakan berikut memungkinkan pengguna untuk menandai sumber daya apa pun yang dapat ditandai, yang sudah memiliki tanda dengan kunci `owner` dan nilai dari nama pengguna. Selain itu, para pengguna juga harus menentukan tanda dengan kunci `anycompany:environment-type` dan nilai dari `test` atau `prod` dalam permintaan. Para pengguna dapat menentukan tanda tambahan dalam permintaan.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
                {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:*/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/anycompany:environment-type":
 ["test","prod"],
                    "aws:ResourceTag/owner": "${aws:username}"
                }
            }
        }
    ]
}
```

Anda dapat membuat kebijakan IAM yang memungkinkan para pengguna untuk menghapus tanda tertentu untuk sumber daya. Sebagai contoh, kebijakan berikut ini memungkinkan para pengguna untuk menghapus tanda untuk volume jika kunci tanda yang ditentukan dalam permintaan tersebut adalah `environment` atau `cost-center`. Nilai apa pun dapat ditentukan untuk tanda tetapi kunci tanda harus cocok dengan salah satu kunci dari kunci yang ditentukan.

> **Note**
>
> Jika Anda menghapus sumber daya, semua tanda yang dikaitkan dengan sumber daya tersebut juga dihapus. Para pengguna tidak memerlukan izin untuk menggunakan tindakan

`ec2:DeleteTags` untuk menghapus sumber daya yang memiliki tanda; mereka hanya memerlukan izin untuk melakukan tindakan penghapusan.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
       {
      "Effect": "Allow",
      "Action": "ec2:DeleteTags",
      "Resource": "arn:aws:ec2:us-east-1:111122223333:volume/*",
      "Condition": {
        "ForAllValues:StringEquals": {
          "aws:TagKeys": ["environment","cost-center"]
        }
      }
    }
  ]
}
```

Kebijakan ini memungkinkan pengguna untuk hanya menghapus tanda `environment=prod` pada sumber daya mana pun, dan hanya jika sumber daya tersebut sudah ditandai dengan kunci `owner` dan nilai dari nama pengguna. Pengguna tidak dapat menghapus tanda lain untuk sumber daya.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
      {
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:*/*",
      "Condition": {
        "StringEquals": {
```

```
          "aws:RequestTag/environment": "prod",
          "aws:ResourceTag/owner": "${aws:username}"
        },
        "ForAllValues:StringEquals": {
          "aws:TagKeys": ["environment"]
        }
      }
    }
  ]
}
```

## Contoh: Cara Menggunakan IAM role

Kebijakan berikut ini memungkinkan para pengguna untuk melampirkan, mengganti, dan melepaskan IAM role ke instans yang memiliki tanda `department=test`. Mengganti atau melepaskan IAM role terlampir memerlukan ID asosiasi, oleh karena itu kebijakan tersebut juga memberikan izin kepada para pengguna untuk menggunakan tindakan `ec2:DescribeIamInstanceProfileAssociations`.

Pengguna harus memiliki izin untuk menggunakan `iam:PassRole` tindakan untuk meneruskan peran ke instance.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:AssociateIamInstanceProfile",
        "ec2:ReplaceIamInstanceProfileAssociation",
        "ec2:DisassociateIamInstanceProfile"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/department":"test"
        }
      }
    },
```

```
    {
      "Effect": "Allow",
      "Action": "ec2:DescribeIamInstanceProfileAssociations",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::111122223333:role/DevTeam*"
    }
  ]
}
```

Kebijakan berikut ini akan memungkinkan para pengguna untuk melampirkan atau mengganti IAM role untuk instans apa pun. Para pengguna hanya dapat melampirkan atau mengganti IAM role dengan nama yang dimulai dengan `TestRole-`. Untuk tindakan `iam:PassRole`, pastikan Anda mencantumkan nama IAM role dan bukan nama profil instans (jika nama keduanya berbeda). Untuk informasi selengkapnya, lihat Profil instans.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AssociateIamInstanceProfile",
                "ec2:ReplaceIamInstanceProfileAssociation"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:DescribeIamInstanceProfileAssociations",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/TestRole-*"
```

```
        }
    ]
}
```

## Contoh: Cara menggunakan tabel rute

Kebijakan berikut ini memungkinkan para pengguna menambahkan, menghapus, dan mengganti rute untuk tabel rute yang dikaitkan dengan VPC `vpc-ec43eb89` saja. Untuk menentukan VPC untuk kunci syarat `ec2:Vpc`, Anda harus menentukan ARN lengkap dari VPC tersebut.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
              {
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteRoute",
                "ec2:CreateRoute",
                "ec2:ReplaceRoute"
            ],
            "Resource": [
                "arn:aws:ec2:us-east-1:111122223333:route-table/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:Vpc": "arn:aws:ec2:us-east-1:111122223333:vpc/vpc-
ec43eb89"
                }
            }
        }
    ]
}
```

## Contoh: Mengizinkan instance tertentu untuk melihat sumber daya di lainnya AWS layanan

Berikut ini adalah contoh kebijakan yang dapat Anda lampirkan ke IAM role. Kebijakan ini memungkinkan instance untuk melihat sumber daya di berbagai AWS layanan. Ini menggunakan kunci kondisi `ec2:SourceInstanceARN` global untuk menentukan bahwa instance dari mana permintaan dibuat harus berupa instance`i-093452212644b0dd6`. Jika IAM role yang sama dikaitkan dengan instans yang lain, maka instans lain tersebut tidak akan dapat melakukan tindakan apa pun.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
              {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeVolumes",
                "s3:ListAllMyBuckets",
                "dynamodb:ListTables",
                "rds:DescribeDBInstances"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "ArnEquals": {
                    "ec2:SourceInstanceARN": "arn:aws:ec2:us-
east-1:111122223333:instance/i-093452212644b0dd6"
                }
            }
        }
    ]
}
```

## Contoh: Cara menggunakan templat peluncuran

Kebijakan berikut ini memungkinkan para pengguna untuk membuat versi templat peluncuran dan memodifikasi templat peluncuran, tetapi hanya untuk templat peluncuran tertentu

(`lt-09477bcd97b0d3abc`). Para pengguna tidak dapat menggunakan templat peluncuran yang lain.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
   {
      "Action": [
        "ec2:CreateLaunchTemplateVersion",
        "ec2:ModifyLaunchTemplate"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:us-east-1:111122223333:launch-template/
lt-09477bcd97b0d3abc"
    }
  ]
}
```

Kebijakan berikut ini akan memungkinkan para pengguna untuk menghapus templat peluncuran dan versi templat peluncuran, dengan ketentuan bahwa templat peluncuran tersebut memiliki tanda `Purpose=Testing`.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
       {
      "Action": [
        "ec2:DeleteLaunchTemplate",
        "ec2:DeleteLaunchTemplateVersions"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:ec2:us-east-1:111122223333:launch-template/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Purpose": "Testing"
        }
```

```
        }
      }
    ]
}
```

## Cara menggunakan metadata instans

Kebijakan berikut ini akan memastikan bahwa para pengguna hanya dapat mengambil metadata instans menggunakan Instans Metadata Service Version 2 (IMDSv2). Anda dapat menggabungkan empat kebijakan berikut ini ke dalam satu kebijakan yang memiliki empat pernyataan. Jika digabungkan sebagai satu kebijakan, Anda dapat menggunakan kebijakan tersebut sebagai kebijakan kontrol layanan (SCP). Kebijakan tersebut dapat berfungsi dengan sama baiknya sebagai kebijakan tolak yang Anda terapkan pada kebijakan IAM yang sudah ada (menarik dan membatasi izin dengan sudah ada), atau sebagai SCP yang diterapkan secara global pada akun, unit organisasi (OU), atau keseluruhan organisasi.

> **Note**
>
> Kebijakan opsi RunInstances metadata berikut harus digunakan bersama dengan kebijakan yang memberikan izin utama untuk meluncurkan instance dengan RunInstances. Jika kepala sekolah juga tidak memiliki RunInstances izin, ia tidak akan dapat meluncurkan instance. Untuk informasi selengkapnya, lihat kebijakan-kebijakan yang ada dalam Cara menggunakan instans dan Luncurkan instance (RunInstances).

> **Important**
>
> Jika Anda menggunakan grup Auto Scaling dan Anda membutuhkan penggunaan IMDSv2 pada semua instans baru, maka grup Auto Scaling Anda harus menggunakan templat peluncuran.
> Saat grup Auto Scaling menggunakan templat peluncuran, izin `ec2:RunInstances` dari prinsipal utama IAM akan diperiksa saat grup Auto Scaling baru dibuat. Izin tersebut juga akan diperiksa saat grup Auto Scaling yang sudah ada diperbarui untuk penggunaan templat peluncuran baru atau templat peluncuran versi baru.
> Pembatasan penggunaan IMDSv1 pada prinsipal utama IAM untuk `RunInstances` hanya akan diperiksa saat ada grup Auto Scaling yang menggunakan templat peluncuran, dibuat atau diperbarui. Untuk grup Auto Scaling yang dikonfigurasi untuk menggunakan templat

peluncuran `Latest` atau `Default`, izin tersebut tidak diperiksa saat versi baru dari templat peluncuran tersebut dibuat. Untuk izin yang akan diperiksa, pengguna harus melakukan konfigurasi terhadap grup Auto Scaling untuk menggunakan versi tertentu dari templat peluncuran tersebut.

Untuk menerapkan penggunaan IMDSv2 pada instans yang diluncurkan oleh grup Auto Scaling, perlu dilakukan langkah-langkah tambahan berikut ini:

1. Nonaktifkan penggunaan konfigurasi peluncuran untuk semua akun dalam organisasi Anda menggunakan kebijakan kontrol layanan (SCP) atau batas-batas izin IAM untuk prinsipal utama baru yang dibuat. Untuk prinsipal utama IAM yang sudah ada yang memiliki izin grup Auto Scaling, lakukan pembaruan atas kebijakan-kebijakan yang dikaitkan dengan kunci syarat ini. Untuk menonaktifkan penggunaan konfigurasi peluncuran, buatlah atau lakukan modifikasi pada SCP, batas-batas izin, atau kebijakan IAM yang relevan dengan kunci syarat `"autoscaling:LaunchConfigurationName"` dengan nilai yang ditentukan sebagai `null`.
2. Untuk templat peluncuran baru, lakukan konfigurasi pada opsi metadata instans di templat peluncuran. Untuk templat peluncuran yang sudah ada, buatlah templat peluncuran versi baru dan lakukan konfigurasi pada opsi metadata instans dalam versi baru itu.
3. Dalam kebijakan yang memberikan izin kepada setiap prinsipal utama untuk menggunakan templat peluncuran, batasi asosiasi `$latest` dan `$default` dengan menentukan `"autoscaling:LaunchTemplateVersionSpecified": "true"`. Dengan membatasi penggunaan hanya pada templat peluncuran versi tertentu saja, Anda telah memastikan bahwa instans baru akan diluncurkan menggunakan versi di mana opsi metadata dikonfigurasi. Untuk informasi selengkapnya, lihat LaunchTemplateSpecification dalam Referensi Amazon EC2 Auto Scaling API, khususnya parameter `Version`.
4. Untuk grup Auto Scaling yang menggunakan konfigurasi peluncuran, ganti konfigurasi peluncuran itu dengan templat peluncuran. Untuk informasi selengkapnya, lihat Mem igrasi grup Penskalaan Otomatis Anda untuk meluncurkan templat di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
5. Untuk grup Auto Scaling yang menggunakan templat peluncuran, pastikan grup tersebut menggunakan templat peluncuran baru dengan opsi metadata instans yang telah dikonfigurasi, atau menggunakan templat peluncuran versi terbaru saat ini dengan opsi metadata instans yang telah dikonfigurasi. Untuk informasi selengkapnya, lihat update-auto-scaling-group.

Contoh

- Mengharuskan penggunaan IMDSv2
- Tolak opt-out dari IMDSv2
- Menentukan batas hop maksimum
- Batasi siapa saja yang dapat melakukan modifikasi terhadap opsi metadata instans
- Mengharuskan kredensial peran untuk diambil dari IMDSv2

Mengharuskan penggunaan IMDSv2

Kebijakan berikut menetapkan bahwa Anda tidak dapat memanggil RunInstances API kecuali instans juga dipilih untuk mewajibkan penggunaan IMDSv2 (ditunjukkan oleh). `"ec2:MetadataHttpTokens": "required"` Jika Anda tidak menentukan bahwa instance memerlukan IMDSv2, Anda mendapatkan `UnauthorizedOperation` kesalahan saat memanggil API. RunInstances

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
                {
            "Sid": "RequireImdsV2",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
                "StringNotEquals": {
                    "ec2:MetadataHttpTokens": "required"
                }
            }
        }
    ]
}
```

Tolak opt-out dari IMDSv2

Kebijakan berikut menetapkan bahwa Anda tidak dapat memanggil API `ModifyInstanceMetadataOptions` dan mengizinkan opsi IMDSv1 atau IMDSv2. Jika Anda

memanggil API `ModifyInstanceMetadataOptions`, atribut `HttpTokens` harus diatur ke `required`.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
        "Sid": "DenyIMDSv1HttpTokensModification",
        "Effect": "Deny",
        "Action": "ec2:ModifyInstanceMetadataOptions",
        "Resource": "arn:aws:ec2:*:*:instance/*",
        "Condition": {
            "StringNotEquals": {
                "ec2:Attribute/HttpTokens": "required"
            },
            "Null": {
                "ec2:Attribute/HttpTokens": false
            }
        }
    }]
}
```

Menentukan batas hop maksimum

Kebijakan berikut menetapkan bahwa Anda tidak dapat memanggil RunInstances API kecuali Anda juga menentukan batas hop, dan batas hop tidak boleh lebih dari 3. Jika Anda gagal melakukannya, Anda mendapatkan `UnauthorizedOperation` kesalahan saat memanggil RunInstances API.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
                {
            "Sid": "MaxImdsHopLimit",
            "Effect": "Deny",
            "Action": "ec2:RunInstances",
            "Resource": "arn:aws:ec2:*:*:instance/*",
            "Condition": {
```

```
                "NumericGreaterThan": {
                    "ec2:MetadataHttpPutResponseHopLimit": "3"
                }
            }
        }
    ]
}
```

Batasi siapa saja yang dapat melakukan modifikasi terhadap opsi metadata instans

Kebijakan berikut hanya mengizinkan pengguna dengan peran `ec2-imds-admins` untuk melakukan perubahan pada opsi metadata instans. Jika ada prinsipal selain peran `ec2-imds-admins` mencoba untuk memerintahkan ModifyInstanceMetadataOptions API, eror `Unauthorized0peration` akan terjadi. Pernyataan ini dapat digunakan untuk mengontrol penggunaan ModifyInstanceMetadataOptions API; saat ini tidak ada kontrol akses (kondisi) berbutir halus untuk API. ModifyInstanceMetadataOptions

Mengharuskan kredensial peran untuk diambil dari IMDSv2

Kebijakan berikut ini menentukan bahwa jika kebijakan ini diterapkan pada peran, serta peran tersebut diambil oleh layanan EC2 dan kredensial yang dihasilkan digunakan untuk memberikan tanda pada permintaan, maka permintaan tersebut harus ditandatangani oleh kredensial peran EC2 yang diambil dari IMDSv2. Jika tidak, semua panggilan API akan mendapatkan kesalahan `Unauthorized0peration`. Ini statement/policy dapat diterapkan secara umum karena, jika permintaan tidak ditandatangani oleh kredenSIAL peran EC2, itu tidak berpengaruh.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
                {
            "Sid": "RequireAllEc2RolesToUseV2",
            "Effect": "Deny",
            "Action": "*",
            "Resource": "*",
            "Condition": {
                "NumericLessThan": {
                    "ec2:RoleDelivery": "2.0"
                }
```

```
            }
        }
    ]
}
```

### Bekerja dengan volume dan snapshot Amazon EBS

Misalnya kebijakan untuk bekerja dengan volume dan snapshot Amazon EBS, lihat contoh Identity-based kebijakan untuk Amazon EBS.

## Contoh kebijakan untuk mengontrol akses ke konsol Amazon EC2

Anda dapat menggunakan kebijakan IAM untuk memberi pengguna izin yang diperlukan untuk bekerja dengan Amazon EC2. Untuk petunjuk langkah demi langkah, lihat Membuat kebijakan IAM di Panduan Pengguna IAM.

Konsol tersebut menggunakan tindakan-tindakan API tambahan untuk fitur-fiturnya, sehingga kebijakan-kebijakan ini mungkin tidak berjalan sesuai yang diharapkan. Sebagai contoh, seorang pengguna yang memiliki izin untuk hanya menggunakan tindakan API `DescribeVolumes` akan mendapati kesalahan saat mencoba melihat volume dalam konsol. Bagian ini akan menunjukkan kebijakan-kebijakan yang memungkinkan para pengguna untuk menggunakan bagian tertentu dari konsol. Untuk informasi tambahan tentang membuat kebijakan untuk konsol Amazon EC2, lihat posting Blog AWS Keamanan berikut: Memberikan Izin Pengguna untuk Bekerja di Konsol Amazon EC2.

Contoh berikut menunjukkan pernyataan kebijakan yang dapat Anda gunakan untuk memberikan izin kepada pengguna untuk menggunakan Amazon EC2. Ganti masing-masing *`user input placeholder`* dengan informasi Anda sendiri. Kebijakan ini dirancang untuk permintaan yang dibuat menggunakan Konsol Manajemen AWS. Konsol Amazon EC2 mungkin memanggil beberapa tindakan API untuk menampilkan satu sumber daya, dan mungkin tidak jelas sampai pengguna mencoba tugas dan konsol menampilkan kesalahan. Untuk informasi selengkapnya, lihat posting Blog AWS Keamanan berikut: Memberikan Izin Pengguna untuk Bekerja di Amazon EC2 Console.

Contoh

- Contoh: Read-only akses
- Contoh: Menggunakan wizard peluncuran instans EC2
- Contoh: Cara menggunakan grup keamanan

- Contoh: Cara menggunakan alamat IP Elastis
- Contoh: Cara Menggunakan Instans Cadangan

Untuk membantu Anda mengetahui tindakan API mana yang diperlukan untuk melakukan tugas di konsol, Anda dapat menggunakan layanan yang mencatat panggilan, seperti AWS CloudTrail. Jika kebijakan Anda tidak memberikan izin untuk membuat atau melakukan modifikasi terhadap sumber daya tertentu, maka konsol akan menampilkan pesan berenkode yang memuat informasi diagnostik. Anda dapat memecahkan kode pesan menggunakan tindakan DecodeAuthorizationMessage API untuk AWS STS, atau perintah decode-author ization-message di. AWS CLI

## Contoh: Read-only akses

Untuk memungkinkan para pengguna menampilkan semua sumber daya dalam konsol Amazon EC2, Anda dapat menggunakan kebijakan yang sama seperti contoh berikut ini: Contoh: Read-only akses. Para pengguna tidak dapat melakukan tindakan apa pun pada sumber daya tersebut atau membuat sumber daya baru, kecuali bila ada pernyataan lain yang memberikan izin kepada mereka untuk melakukan hal itu.

Tampilkan instans, AMI, dan snapshot

Atau, Anda dapat memberikan akses hanya-baca ke subset sumber daya. Untuk melakukan hal ini, ganti wildcard * dalam tindakan API `ec2:Describe` dengan tindakan `ec2:Describe` tertentu untuk masing-masing sumber daya. Kebijakan berikut ini akan memungkinkan para pengguna untuk menampilkan semua instans, AMI, dan snapshot dalam konsol Amazon EC2. Tindakan `ec2:DescribeTags` akan memungkinkan para pengguna untuk melihat AMI publik. Konsol tersebut membutuhkan informasi penandaan untuk menampilkan AMI publik; akan tetapi, Anda dapat membuang tindakan ini agar pengguna hanya dapat melihat AMI privat saja.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeInstances",
         "ec2:DescribeImages",
         "ec2:DescribeTags",
```

```
            "ec2:DescribeSnapshots"
         ],
         "Resource": "*"
      }
      ]
}
```

> **Note**
>
> Tindakan API `ec2:Describe*` Amazon EC2 tidak mendukung izin tingkat sumber daya, sehingga Anda tidak dapat mengontrol sumber daya individu mana yang dapat dilihat oleh pengguna dalam konsol. Oleh karena itu, wildcard * dibutuhkan dalam elemen `Resource` pada pernyataan di atas. Untuk informasi selengkapnya tentang ARN mana yang dapat Anda gunakan bersama dengan tindakan API Amazon EC2 yang mana, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2.

Melihat instance dan CloudWatch metrik

Kebijakan berikut memungkinkan pengguna untuk melihat instans dalam konsol Amazon EC2, serta alarm dan metrik CloudWatch dalam tab Monitoring dari halaman Instances. Konsol Amazon EC2 menggunakan CloudWatch API untuk menampilkan alarm dan metrik, jadi Anda harus memberikan izin kepada pengguna untuk menggunakan`cloudwatch:DescribeAlarms`,,`cloudwatch:DescribeAlarmsForMetric`, `cloudwatch:ListMetricscloudwatch:GetMetricStatistics`, dan `cloudwatch:GetMetricData` tindakan.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeInstances",
         "ec2:DescribeInstanceTypes",
         "cloudwatch:DescribeAlarms",
         "cloudwatch:DescribeAlarmsForMetric",
         "cloudwatch:ListMetrics",
```

```
            "cloudwatch:GetMetricStatistics",
            "cloudwatch:GetMetricData"
        ],
        "Resource": "*"
    }
    ]
}
```

## Contoh: Menggunakan wizard peluncuran instans EC2

Wizard peluncuran instans Amazon EC2 adalah layar yang menampilkan opsi untuk melakukan konfigurasi dan meluncurkan instans. Kebijakan Anda harus menyertakan izin untuk menggunakan tindakan API yang memungkinkan para pengguna untuk menggunakan opsi-opsi yang ditampilkan pemandu. Jika kebijakan Anda tidak menyertakan izin untuk menggunakan tindakan tersebut, beberapa item dalam pemandu tidak akan dapat dimuat dengan benar, dan pengguna tidak akan dapat menyelesaikan peluncuran.

Akses wizard peluncuran instans dasar

Agar berhasil menyelesaikan peluncuran, para pengguna harus diberi izin untuk menggunakan tindakan API `ec2:RunInstances`, dan setidaknya tindakan-tindakan API berikut ini:

- `ec2:DescribeImages`: Untuk menampilkan dan memilih AMI.
- `ec2:DescribeInstanceTypes`: Untuk menampilkan dan memilih tipe instans.
- `ec2:DescribeVpcs`: Untuk menampilkan opsi-opsi jaringan yang tersedia.
- `ec2:DescribeSubnets`: Untuk menampilkan semua subnet yang tersedia untuk VPC yang dipilih.
- `ec2:DescribeSecurityGroups` atau `ec2:CreateSecurityGroup`: Untuk menampilkan dan memilih grup keamanan yang sudah ada, atau untuk membuat grup keamanan yang baru.
- `ec2:DescribeKeyPairs` atau `ec2:CreateKeyPair`: Untuk memilih pasangan kunci yang sudah ada, atau untuk membuat pasangan kunci yang baru.
- `ec2:AuthorizeSecurityGroupIngress`: Untuk menambahkan aturan ke dalam.

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeImages",
                "ec2:DescribeInstanceTypes",
                "ec2:DescribeKeyPairs",
                "ec2:DescribeVpcs",
                "ec2:DescribeSubnets",
                "ec2:DescribeSecurityGroups",
                "ec2:CreateSecurityGroup",
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:CreateKeyPair"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:RunInstances",
            "Resource": "*"
        }
    ]
}
```

Anda dapat menambahkan tindakan-tindakan API pada kebijakan Anda untuk menyediakan lebih banyak opsi bagi para pengguna, sebagai contoh:

- `ec2:DescribeAvailabilityZones`: Untuk menampilkan dan memilih Zona Ketersediaan tertentu.
- `ec2:DescribeNetworkInterfaces`: Untuk menampilkan dan memilih antarmuka jaringan yang sudah ada untuk subnet yang dipilih.
- Untuk menambahkan aturan-aturan ke luar untuk grup keamanan VPC, para pengguna harus mendapatkan izin untuk menggunakan tindakan API `ec2:AuthorizeSecurityGroupEgress`. Untuk melakukan modifikasi atau menghapus aturan-aturan yang sudah ada, para pengguna harus diberi izin untuk menggunakan tindakan API `ec2:RevokeSecurityGroup*` yang relevan.
- `ec2:CreateTags`: Untuk memberikan tanda pada sumber daya yang dibuat oleh `RunInstances`. Untuk informasi selengkapnya, lihat Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan. Jika pengguna tidak memiliki izin untuk menggunakan tindakan

ini dan mereka berusaha untuk menerapkan tanda di halaman penandaan wizard peluncuran instans, maka peluncuran akan gagal.

> **Important**
>
> Menentukan Nama saat meluncurkan instans membuat tanda dan memerlukan tindakan `ec2:CreateTags`. Anda harus berhati-hati dalam memberikan izin kepada para pengguna untuk menggunakan tindakan `ec2:CreateTags`, karena tindakan itu akan membatasi kemampuan Anda untuk menggunakan kunci syarat `aws:ResourceTag` untuk membatasi penggunaan sumber daya yang lain. Jika Anda memberi pengguna izin untuk menggunakan `ec2:CreateTags` tindakan, mereka dapat mengubah tag sumber daya untuk melewati batasan tersebut. Untuk informasi selengkapnya, lihat Kontrol akses menggunakan akses berbasis atribut.

- Untuk menggunakan parameter Systems Manager saat Anda memilih AMI, Anda harus menambahkan `ssm:DescribeParameters` dan `ssm:GetParameters` pada kebijakan. `ssm:DescribeParameters` memberikan izin kepada pengguna untuk melihat dan memilih parameter Systems Manager. `ssm:GetParameters` memberikan izin kepada pengguna untuk mendapatkan nilai dari parameter Systems Manager. Anda juga dapat membatasi akses ke parameter Systems Manager tertentu. Untuk informasi selengkapnya, lihat Membatasi akses ke parameter Systems Manager tertentu yang juga ada dalam bagian ini.

Saat ini, tindakan API `Describe*` Amazon EC2 tidak mendukung izin tingkat sumber daya, sehingga Anda tidak dapat membatasi sumber daya individu mana yang dapat dilihat dalam wizard peluncuran instans. Akan tetapi, Anda dapat menerapkan izin tingkat sumber daya pada tindakan API `ec2:RunInstances` untuk membatasi sumber daya mana yang dapat digunakan oleh para pengguna untuk meluncurkan instans. Peluncuran tersebut akan gagal jika pengguna memilih opsi-opsi yang tidak mendapatkan otorisasi untuk digunakan.

Membatasi akses ke tipe instans, subnet, dan Wilayah tertentu

Kebijakan berikut ini memungkinkan para pengguna untuk meluncurkan instans `t2.micro` menggunakan AMI yang dimiliki oleh Amazon, dan hanya ke dalam subnet tertentu (`subnet-1a2b3c4d`). Pengguna hanya dapat meluncurkan di Wilayah yang ditentukan. Jika pengguna memilih Wilayah yang berbeda, atau memilih tipe instans, AMI, atau subnet yang berbeda dalam wizard peluncuran instans, maka peluncuran akan gagal.

Pernyataan pertama memberikan izin kepada pengguna untuk melihat opsi dalam wizard peluncuran instans atau untuk membuat yang baru, sebagaimana yang telah dijelaskan dalam contoh di atas. Pernyataan kedua memberikan izin kepada para pengguna untuk menggunakan antarmuka jaringan, volume, pasangan kunci, grup keamanan, dan sumber daya subnet untuk tindakan `ec2:RunInstances`, yang diperlukan untuk meluncurkan instans ke dalam VPC. Untuk informasi selengkapnya tentang penggunaan tindakan `ec2:RunInstances`, lihat Luncurkan instance (RunInstances). Pernyataan ketiga dan keempat memberikan izin kepada pengguna untuk menggunakan masing-masing instans dan sumber daya AMI, tetapi hanya jika instans tersebut adalah instans `t2.micro`, serta hanya jika AMI tersebut dimiliki oleh Amazon atau partner tertentu yang tepercaya dan terverifikasi.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeInstances",
         "ec2:DescribeImages",
         "ec2:DescribeInstanceTypes",
         "ec2:DescribeKeyPairs",
         "ec2:CreateKeyPair",
         "ec2:DescribeVpcs",
         "ec2:DescribeSubnets", "ec2:DescribeSecurityGroups",
         "ec2:CreateSecurityGroup",
         "ec2:AuthorizeSecurityGroupIngress"
   ],
   "Resource": "*"
   },
   {
      "Effect": "Allow",
      "Action":"ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-2:111122223333:network-interface/*",
         "arn:aws:ec2:us-east-2:111122223333:volume/*",
         "arn:aws:ec2:us-east-2:111122223333:key-pair/*",
         "arn:aws:ec2:us-east-2:111122223333:security-group/*",
         "arn:aws:ec2:us-east-2:111122223333:subnet/subnet-1a2b3c4d"
      ]
   },
```

```
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
         "arn:aws:ec2:us-east-2:111122223333:instance/*"
      ],
      "Condition": {
         "StringEquals": {
            "ec2:InstanceType": "t2.micro"
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": "ec2:RunInstances",
      "Resource": [
            "arn:aws:ec2:us-east-2::image/ami-*"
      ],
      "Condition": {
         "StringEquals": {
            "ec2:Owner": "amazon"
         }
      }
   }
   ]
}
```

Membatasi akses ke parameter Systems Manager tertentu

Kebijakan berikut ini memberikan akses untuk menggunakan parameter-parameter Systems Manager yang memiliki nama tertentu.

Pernyataan pertama memberikan izin kepada pengguna untuk menampilkan parameter Systems Manager saat memilih AMI dalam wizard peluncuran instans. Pernyataan kedua memberikan izin kepada para pengguna untuk menggunakan parameter yang mempunyai nama `prod-*`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [{
```

```
      "Effect": "Allow",
      "Action": [
         "ssm:DescribeParameters"
      ],
      "Resource": "*"
   },
   {
      "Effect": "Allow",
      "Action": [
         "ssm:GetParameters"
      ],
     "Resource": "arn:aws:ssm:us-east-2:123456123456:parameter/prod-*"
   }
   ]
}
```

## Contoh: Cara menggunakan grup keamanan

Menampilkan grup keamanan dan menambah serta menghapus aturan

Kebijakan berikut ini memberikan izin kepada para pengguna untuk menampilkan grup keamanan di konsol Amazon EC2, untuk menambahkan dan menghapus aturan ke dalam dan ke luar, dan untuk mencantumkan serta melakukan modifikasi terhadap deskripsi aturan untuk grup keamanan yang sudah ada yang memiliki tanda `Department=Test`.

Dalam pernyataan pertama, tindakan `ec2:DescribeTags` akan memungkinkan para pengguna untuk menampilkan tanda dalam konsol, yang dapat mempermudah para pengguna untuk mengidentifikasi grup keamanan yang diizinkan untuk dimodifikasi.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeSecurityGroups",
         "ec2:DescribeSecurityGroupRules",
         "ec2:DescribeTags"
      ],
      "Resource": "*"
```

```
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:AuthorizeSecurityGroupIngress",
         "ec2:RevokeSecurityGroupIngress",
         "ec2:AuthorizeSecurityGroupEgress",
         "ec2:RevokeSecurityGroupEgress",
         "ec2:ModifySecurityGroupRules",
         "ec2:UpdateSecurityGroupRuleDescriptionsIngress",
         "ec2:UpdateSecurityGroupRuleDescriptionsEgress"
      ],
      "Resource": [
         "arn:aws:ec2:us-east-2:111122223333:security-group/*"
      ],
      "Condition": {
         "StringEquals": {
            "aws:ResourceTag/Department": "Test"
         }
      }
   },
   {
      "Effect": "Allow",
      "Action": [
         "ec2:ModifySecurityGroupRules"
      ],
      "Resource": [
         "arn:aws:ec2:us-east-2:111122223333:security-group-rule/*"
      ]
   }
]}
```

Cara menggunakan kotak dialog Buat Grup Keamanan

Anda dapat membuat kebijakan yang memungkinkan para pengguna untuk menggunakan kotak dialog Buat Grup Keamanan yang ada dalam konsol Amazon EC2. Untuk menggunakan kotak dialog ini, para pengguna harus diberi izin untuk menggunakan setidaknya tindakan-tindakan API berikut ini:

- `ec2:CreateSecurityGroup`: Untuk membuat grup keamanan yang baru.
- `ec2:DescribeVpcs`: Untuk menampilkan daftar VPC yang sudah ada dalam daftar VPC.

Dengan izin tersebut, para pengguna dapat membuat grup keamanan baru dengan sukses, tetapi mereka tidak akan dapat menambahkan aturan apa pun pada grup keamanan tersebut. Untuk menggunakan aturan-aturan yang ada dalam kotak dialog Buat Grup Keamanan, Anda dapat menambahkan tindakan-tindakan API berikut pada kebijakan Anda:

- `ec2:AuthorizeSecurityGroupIngress`: Untuk menambahkan aturan ke dalam.
- `ec2:AuthorizeSecurityGroupEgress`: Untuk menambahkan aturan ke luar pada grup keamanan VPC.
- `ec2:RevokeSecurityGroupIngress`: Untuk melakukan modifikasi atau membuang aturan ke dalam yang sudah ada. Tindakan-tindakan ini berguna untuk memungkinkan para pengguna menggunakan fitur Salin ke yang baru yang ada dalam konsol. Fitur ini akan membuka kotak dialog Buat Grup Keamanan dan mengisinya dengan aturan-aturan yang sama seperti grup keamanan yang sudah dipilih.
- `ec2:RevokeSecurityGroupEgress`: Untuk melakukan modifikasi atau penghapusan terhadap aturan-aturan ke luar untuk grup keamanan VPC. Hal ini berguna untuk memungkinkan para pengguna untuk melakukan modifikasi terhadap atau menghapus aturan ke luar default yang mengizinkan semua lalu lintas ke luar.
- `ec2:DeleteSecurityGroup`: Untuk melayani ketika aturan-aturan yang tidak valid tidak dapat disimpan. Pertama-tama konsol akan membuat grup keamanan, kemudian akan menambahkan aturan-aturan tertentu. Jika aturan tidak valid, maka tindakan tersebut akan gagal, dan konsol akan mencoba menghapus grup keamanan. Para pengguna akan tetap berada dalam kotak dialog Buat Grup Keamanan sehingga mereka dapat melakukan koreksi atas aturan-aturan yang tidak valid dan mencoba membuat grup keamanan lagi. Tindakan API ini tidak diperlukan, tetapi jika seorang pengguna tidak diberikan izin untuk menggunakannya dan berusaha untuk membuat grup keamanan dengan aturan-aturan yang tidak valid, maka grup keamanan akan dibuat tanpa aturan apa pun, dan pengguna tersebut harus menambahkan aturan-aturan setelahnya.
- `ec2:UpdateSecurityGroupRuleDescriptionsIngress`: Untuk menambahkan atau memperbarui deskripsi aturan grup keamanan ingress (ke dalam).
- `ec2:UpdateSecurityGroupRuleDescriptionsEgress`: Untuk menambahkan atau memperbarui deskripsi aturan grup keamanan egress (ke luar).
- `ec2:ModifySecurityGroupRules`: Untuk mengubah aturan grup keamanan.
- `ec2:DescribeSecurityGroupRules`: Untuk mencantumkan aturan grup keamanan.

Kebijakan berikut ini akan memberikan izin kepada para pengguna untuk menggunakan kotak dialog Buat Grup Keamanan dan untuk membuat aturan-aturan ke dalam dan ke luar untuk grup keamanan

yang dikaitkan dengan VPC tertentu (`vpc-1a2b3c4d`). Pengguna dapat membuat grup keamanan untuk VPC, tetapi mereka tidak dapat menambahkan aturan apa pun pada grup keamanan tersebut. Demikian pula, para pengguna tidak dapat menambahkan aturan apa pun ke grup keamanan yang ada yang tidak dikaitkan dengan VPC `vpc-1a2b3c4d`. Para pengguna juga diberikan izin untuk menampilkan semua grup keamanan di konsol. Hal ini akan mempermudah para pengguna untuk mengidentifikasi grup keamanan yang padanya dapat mereka tambahkan aturan-aturan ke dalam. Kebijakan ini juga memberikan izin kepada para pengguna untuk menghapus grup keamanan yang dikaitkan dengan VPC `vpc-1a2b3c4d`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeSecurityGroups",
        "ec2:CreateSecurityGroup",
        "ec2:DescribeVpcs"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:DeleteSecurityGroup",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:AuthorizeSecurityGroupEgress"
      ],
      "Resource": "arn:aws:ec2:us-east-2:111122223333:security-group/*",
      "Condition":{
         "ArnEquals": {
            "ec2:Vpc": "arn:aws:ec2:us-east-2:111122223333:vpc/vpc-1a2b3c4d"
         }
      }
    }
   ]
}
```

## Contoh: Cara menggunakan alamat IP Elastis

Untuk memungkinkan para pengguna menampilkan alamat IP Elastis dalam konsol Amazon EC2, Anda harus memberikan izin kepada para pengguna untuk menggunakan tindakan `ec2:DescribeAddresses`.

Agar pengguna dapat menggunakan alamat IP Elastis, Anda dapat menambahkan tindakan-tindakan berikut pada kebijakan Anda.

- `ec2:AllocateAddress`: Untuk mengalokasikan alamat IP Elastis.
- `ec2:ReleaseAddress`: Untuk merilis alamat IP Elastis.
- `ec2:AssociateAddress`: Untuk mengaitkan alamat IP Elastis dengan instans atau antarmuka jaringan.
- `ec2:DescribeNetworkInterfaces` dan `ec2:DescribeInstances`: Untuk menggunakan layar Kaitkan alamat. Layar tersebut akan menampilkan instans atau antarmuka jaringan yang tersedia yang bisa Anda gunakan untuk mengaitkan alamat IP Elastis.
- `ec2:DisassociateAddress`: Untuk melepaskan pengaitan alamat IP Elastis dari instans atau antarmuka jaringan.

Kebijakan berikut ini akan memungkinkan para pengguna untuk menampilkan, mengalokasikan, dan mengaitkan alamat IP Elastis dengan instans. Para pengguna tidak dapat mengaitkan alamat IP Elastis dengan antarmuka jaringan, melepaskan pengaitan alamat IP Elastis, atau merilisnya.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeAddresses",
                "ec2:AllocateAddress",
                "ec2:DescribeInstances",
                "ec2:AssociateAddress"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

## Contoh: Cara Menggunakan Instans Cadangan

Kebijakan berikut mengizinkan pengguna untuk menampilkan dan memodifikasi Instans Terpesan dalam akun Anda, serta membeli Instans Terpesan baru dalam Konsol Manajemen AWS.

Kebijakan ini memungkinkan pengguna untuk melihat semua Instans Cadangan, serta On-Demand Instans, di akun. Pengaturan izin tingkat sumber daya untuk masing-masing Instans Cadangan tidak dapat dilakukan.

JSON

```
{
   "Version":"2012-10-17",
   "Statement": [{
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeReservedInstances",
         "ec2:ModifyReservedInstances",
         "ec2:PurchaseReservedInstancesOffering",
         "ec2:DescribeInstances",
         "ec2:DescribeInstanceTypes",
         "ec2:DescribeAvailabilityZones",
         "ec2:DescribeReservedInstancesOfferings"
      ],
      "Resource": "*"
   }
   ]
}
```

Tindakan `ec2:DescribeAvailabilityZones` dibutuhkan untuk memastikan bahwa konsol Amazon EC2 dapat menampilkan informasi tentang Zona Ketersediaan di mana Anda dapat membeli Instans Cadangan. Tindakan `ec2:DescribeInstances` tidak diperlukan, tetapi dapat memastikan bahwa pengguna dapat menampilkan instans dalam akun dan membeli cadangan agar sesuai dengan spesifikasi yang semestinya.

Anda dapat menyesuaikan tindakan API untuk membatasi akses pengguna, sebagai contoh, menghapus `ec2:DescribeInstances` dan `ec2:DescribeAvailabilityZones` artinya pengguna memiliki akses hanya-baca.

## AWS kebijakan terkelola untuk Amazon EC2

Untuk menambahkan izin ke pengguna, grup, dan peran, lebih mudah menggunakan kebijakan AWS terkelola daripada menulis kebijakan sendiri. Dibutuhkan waktu dan keahlian untuk membuat kebijakan yang dikelola pelanggan IAM yang hanya memberi tim Anda izin yang mereka butuhkan. Untuk memulai dengan cepat, Anda dapat menggunakan kebijakan terkel AWS ola kami. Kebijakan ini mencakup kasus penggunaan umum dan tersedia di AWS akun Anda. Untuk informasi selengkapnya tentang AWS kebijakan terkelola, AWS lihat kebijakan ter kelola di Panduan Pengguna IAM.

AWS layanan memelihara dan memperbarui kebijakan AWS terkelola. Anda tidak dapat mengubah izin dalam kebijakan ter AWS kelola. Layanan terkadang menambahkan izin tambahan ke kebijakan yang dikelola AWS untuk mendukung fitur-fitur baru. Jenis pembaruan ini akan memengaruhi semua identitas (pengguna, grup, dan peran) di mana kebijakan tersebut dilampirkan. Layanan kemungkinan besar akan memperbarui kebijakan yang dikelola AWS saat ada fitur baru yang diluncurkan atau saat ada operasi baru yang tersedia. Layanan tidak menghapus izin dari kebijakan ter AWS kelola, sehingga pembaruan kebijakan tidak akan merusak izin yang ada.

Selain itu, AWS mendukung kebijakan terkelola untuk fungsi pekerjaan yang mencakup beberapa layanan. Misalnya, kebijakan terkel ReadOnlyAccess AWS ola menyediakan akses baca-saja ke semua AWS layanan dan sumber daya. Saat layanan meluncurkan fitur baru, AWS menambahkan izin baca-saja untuk operasi dan sumber daya baru. Untuk melihat daftar dan deskripsi dari kebijakan fungsi tugas, lihat kebijakan yang dikelola AWS untuk fungsi tugas di Panduan Pengguna IAM.

### AWS kebijakan yang dikelola: AmazonEC2FullAccess

Anda dapat melampirkan kebijakan `AmazonEC2FullAccess` ke identitas-identitas IAM Anda. Kebijakan ini akan memberikan izin yang mengizinkan akses penuh ke Amazon EC2.

Untuk melihat izin kebijakan ini, lihat AmazonEC2FullAccess di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AmazonEC2ReadOnlyAccess

Anda dapat melampirkan kebijakan `AmazonEC2ReadOnlyAccess` ke identitas-identitas IAM Anda. Kebijakan ini memberikan izin yang memungkinkan akses hanya-baca ke Amazon EC2.

Untuk melihat izin kebijakan ini, lihat AmazonEC2ReadOnlyAccess di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AmazonEC2ImageReferencesAccessPolicy

Anda dapat melampirkan kebijakan `AmazonEC2ImageReferencesAccessPolicy` ke identitas IAM Anda. Kebijakan ini memberikan izin yang diperlukan untuk menggunakan DescribeImageReferences API EC2, termasuk izin untuk melihat instans EC2, template peluncuran, parameter Manajer Sistem, dan resep Pembuat Gambar. Kebijakan mendukung `IncludeAllResourceTypes` bendera dan akan terus berfungsi saat AWS menambahkan dukungan untuk jenis sumber daya baru, menghilangkan kebutuhan untuk pembaruan kebijakan di masa mendatang.

Untuk melihat izin kebijakan ini, lihat AmazonEC2ImageReferencesAccessPolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AWSEC2CapacityReservationFleetRolePolicy

Kebijakan ini dilampirkan ke peran terkait layanan yang diberi nama AWSServiceRoleForEC2CapacityReservationFleet untuk memungkinkan layanan membuat, mengubah, dan membatalkan Reservasi Kapasitas di Armada Reservasi Kapasitas atas nama Anda. Untuk informasi selengkapnya, lihat Menggunakan peran tertaut layanan untuk Armada Reservasi Kapasitas.

Untuk melihat izin kebijakan ini, lihat AWSEC2CapacityReservationFleetRolePolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AWSEC2FleetServiceRolePolicy

Kebijakan ini dilampirkan pada peran tertaut layanan yang mempunyai nama AWSServiceRoleForEC2Fleet untuk mengizinkan Armada EC2 meminta, meluncurkan, mengakhiri, dan memberi tanda pada instans atas nama Anda. Untuk informasi selengkapnya, lihat Service-linked peran untuk Armada EC2.

Untuk melihat izin kebijakan ini, lihat AWSEC2FleetServiceRolePolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AWSEC2SpotFleetServiceRolePolicy

Kebijakan ini dilampirkan pada peran tertaut layanan yang mempunyai nama AWSServiceRoleForEC2SpotFleet untuk mengizinkan Armada Spot meluncurkan dan mengelola instans atas nama Anda. Untuk informasi selengkapnya, lihat Service-linked peran untuk Spot Fleet.

Untuk melihat izin kebijakan ini, lihat AWSEC2SpotFleetServiceRolePolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AWSEC2SpotServiceRolePolicy

Kebijakan ini dilampirkan pada peran tertaut layanan yang mempunyai nama AWSServiceRoleForEC2Spot untuk mengizinkan Amazon EC2 meluncurkan dan mengelola Instans Spot atas nama Anda. Untuk informasi selengkapnya, lihat Service-linked peran untuk permintaan Instance Spot.

Untuk melihat izin kebijakan ini, lihat AWSEC2SpotServiceRolePolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AWSEC2VssSnapshotPolicy

Anda dapat melampirkan kebijakan terkelola ini ke peran profil instans IAM yang Anda gunakan untuk instans Windows Amazon EC2 Anda. Kebijakan ini memberikan izin untuk mengizinkan Amazon EC2 membuat dan mengelola snapshot VSS atas nama Anda.

Untuk melihat izin kebijakan ini, lihat AWSEC2VssSnapshotPolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: DeclarativePoliciesEC2Report

Kebijakan ini dilampirkan ke peran terkait layanan yang diberi nama `AWSServiceRoleForDeclarativePoliciesEC2Report` untuk menyediakan akses ke API baca-saja yang diperlukan untuk membuat laporan status akun untuk kebijakan deklaratif.

Untuk melihat izin kebijakan ini, lihat DeclarativePoliciesEC2Report di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: EC2FastLaunchFullAccess

Anda dapat melampirkan `EC2FastLaunchFullAccess` kebijakan ke profil instans atau peran IAM lainnya. Kebijakan ini memberikan akses penuh ke tindakan Peluncuran Cepat EC2, dan izin yang ditargetkan sebagai berikut.

Detail izin

- Peluncuran Cepat EC2 - Akses administratif diberikan, sehingga peran tersebut dapat mengaktifkan atau menonaktifkan Peluncuran Cepat EC2, dan menjelaskan gambar Peluncuran Cepat EC2.

- Amazon EC2 — Akses diberikan untuk operasi Amazon EC2 RunInstances, DeskripsiCreateTags, dan Buat dan Ubah Template Peluncuran. Akses juga diberikan untuk membuat sumber daya jaringan dan grup keamanan, mengotorisasi aturan masuk, dan menghapus sumber daya yang dibuat EC2 Fast Launch.
- IAM — Akses diberikan untuk mendapatkan dan menggunakan profil instance yang namanya berisi `ec2fastlaunch` untuk membuat peran terkait EC2FastLaunchServiceRolePolicy layanan.
- CloudFormation— Akses diberikan untuk EC2 Fast Launch untuk mendeskripsikan dan membuat tumpu CloudFormation kan, dan untuk menghapus tumpukan yang dibuatnya.

Untuk melihat izin kebijakan ini, lihat EC2FastLaunchFullAccess di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: AWSEC2CapacityManagerServiceRolePolicy

Kebijakan ini dilampirkan ke peran terkait layanan yang dinamai AWSServiceRoleForEC2CapacityManager untuk memungkinkan Manajer Kapasitas EC2 mengelola sumber daya kapasitas dan berintegrasi dengan AWS Organisasi atas nama Anda. Untuk informasi selengkapnya, lihat Service-linked peran untuk Manajer Kapasitas EC2.

Untuk melihat izin kebijakan ini, lihat AWSEC2CapacityManagerServiceRolePolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: EC2FastLaunchServiceRolePolicy

Kebijakan ini dilampirkan ke peran terkait layanan yang diberi nama AWSServiceRoleForEC2FastLaunch untuk memungkinkan Amazon EC2 membuat dan mengelola serangkaian snapshot yang telah disediakan sebelumnya yang mengurangi waktu yang dibutuhkan untuk meluncurkan instans dari AMI Cepat EC2 Anda. Launch-enabled Untuk informasi selengkapnya, lihat the section called “Service-linked peran”.

Untuk melihat izin kebijakan ini, lihat EC2FastLaunchServiceRolePolicy di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: Ec2InstanceConnect

Anda dapat melampirkan kebijakan `Ec2InstanceConnect` ke identitas IAM Anda. Kebijakan ini memberikan izin yang memungkinkan pelanggan memanggil EC2 Instance Connect untuk menerbitkan kunci sementara ke instans EC2 mereka dan terhubung melalui SSH atau CLI Connect Instance EC2.

Untuk melihat izin kebijakan ini, lihat Ec2InstanceConnect di Referensi Kebijakan AWS Terkelola.

### AWS kebijakan yang dikelola: Ec2InstanceConnectEndpoint

Kebijakan ini dilampirkan ke peran terkait layanan yang diberi nama AWSServiceRoleForEC2InstanceConnect untuk memungkinkan EC2 Instance Connect Endpoint melakukan tindakan atas nama Anda. Untuk informasi selengkapnya, lihat Service-linked peran untuk EC2 Instance Connect Endpoint.

Untuk melihat izin kebijakan ini, lihat Ec2InstanceConnectEndpoint di Referensi Kebijakan AWS Terkelola. Untuk deskripsi pembaruan kebijakan ini, lihatAmazon EC2 memperbarui ke AWS kebijakan terkelola.

### Amazon EC2 memperbarui ke AWS kebijakan terkelola

Lihat detail tentang pembaruan kebijakan ter AWS kelola untuk Amazon EC2 sejak layanan ini mulai melacak perubahan ini.

| Ubah | Deskripsi | Date |
| --- | --- | --- |
| AWSEC2CapacityManagerServiceRolePolicy – Kebijakan baru | Amazon EC2 menambahkan kebijakan ini untuk memungkinkan Anda mengelola sumber daya kapasitas dan berintegrasi dengan AWS Organisasi atas nama Anda. | Oktober 15, 2025 |
| AmazonEC2ImageReferencesAccessPolicy – Kebijakan baru | Amazon EC2 menambahkan kebijakan ini untuk memberikan izin untuk memindai semua jenis sumber daya yang didukung oleh API EC2 DescribeImageReferences . | Agustus 26, 2025 |
| Ec2InstanceConnectEndpoint— Kebijakan yang diperbarui | Untuk mendukung modifikasi Titik Akhir Instance Connect yang ada, Amazon EC2 memperbarui kebijakan ini untuk menambahkan | Juli 31, 2025 |

| Ubah | Deskripsi | Date |
| --- | --- | --- |
| | izin untuk menetapkan dan membatalkan penetapan alamat IPv6 dan memodifik asi grup keamanan pada antarmuka jaringan yang dibuat oleh EC2 Instance Connect Endpoint. Amazon EC2 juga memperbarui kebijakan ini untuk mengganti operator `Null` kondisi dengan operator `StringLike` kondisi. | |
| EC2FastLaunchServiceRolePolicy— Kebijakan yang diperbarui | Untuk membantu mencegah sumber daya yatim piatu, Amazon EC2 memperbarui kebijakan ini untuk menambahkan izin untuk menjelaskan volume, atribut volume, dan antarmuka jaringan, serta untuk menghapus volume dan antarmuka jaringan yang dibuat EC2 Fast Launch. | Juli 17, 2025 |

| Ubah | Deskripsi | Date |
| --- | --- | --- |
| EC2FastLaunchFullAccess— Kebijakan yang diperbarui | Amazon EC2 memperbarui kebijakan ini untuk menyertakan operasi Buat dan Ubah Template Peluncuran, untuk membuat sumber daya grup jaringan dan keamanan, mengotorisasi aturan masuk, dan menghapus sumber daya yang dibuat EC2 Fast Launch. Ini juga dapat menggambarkan dan membuat tumpukan CloudFormation , dan menghapus tumpukan yang dibuat EC2 Fast Launch. | Mei 14, 2025 |
| EC2FastLaunchServiceRolePolicy— Kebijakan yang diperbarui | Amazon EC2 memperbarui kebijakan ini untuk menambahkan EventBridge akses Amazon untuk membuat dan mengelola aturan acara untuk Peluncuran Cepat EC2. Selain itu, EC2 Fast Launch sekarang CloudFormation dapat menjelaskan tumpukan, meluncurkan instance dari AMI yang terkait dengannya AWS License Manager, mendapatkan daftar AWS KMS hibah yang dibuat yang dapat dihentikan, dan menghapus template peluncuran yang dibuat. | Mei 14, 2025 |

| Ubah | Deskripsi | Date |
| --- | --- | --- |
| AWSEC2CapacityReservationFleetRolePolicy— Izin yang diperbarui | Amazon EC2 memperbarui kebijakan ter `AWSEC2CapacityReservationFleetRolePolicy` kelola untuk menggunakan operator `ArnLike` kondisi alih-alih operator `StringLike` kondisi. | Maret 03, 2025 |
| AmazonEC2ReadOnlyAccess— Menambahkan izin | Amazon EC2 menambahkan izin yang memungkinkan Anda mengambil grup keamanan menggunakan `GetSecurityGroupsForVpc` operasi. | Desember 27, 2024 |
| EC2FastLaunchFullAccess – Kebijakan baru | Amazon EC2 menambahkan kebijakan ini untuk melakukan tindakan API yang terkait dengan fitur Peluncuran Cepat EC2 dari instans. Kebijakan dapat dilampirkan ke profil instans untuk instans yang diluncurkan dari AMI yang diaktifkan EC2 Fast Launch. | Mei 14, 2024 |
| AWSEC2VssSnapshotPolicy – Kebijakan baru | Amazon EC2 menambahkan `AWSEC2VssSnapshotPolicy` kebijakan yang berisi izin untuk membuat dan menambahkan tag ke Amazon Machine Images (AMI) dan snapshot EBS. | Maret 28, 2024 |

| Ubah | Deskripsi | Date |
| --- | --- | --- |
| Ec2InstanceConnectEndpoint – Kebijakan baru | Amazon EC2 menambahkan `Ec2InstanceConnectEndpoint` kebijakan tersebut. Kebijakan ini dilampirkan ke peran AWSServiceRoleForEC2InstanceConnect terkait layanan, untuk memungkinkan Amazon EC2 melakukan tindakan atas nama Anda saat Anda membuat Titik Akhir Connect Instance EC2. | Januari 24, 2023 |
| EC2FastLaunchServiceRolePolicy – Kebijakan baru | Amazon EC2 menambahkan fitur EC2 Fast Launch untuk memungkinkan AMI Windows meluncurkan instans lebih cepat dengan membuat satu set snapshot yang telah disediakan sebelumnya. | 26 November 2021 |
| Amazon EC2 mulai melakukan pelacakan perubahan | Amazon EC2 mulai melacak perubahan pada kebijakan yang AWS dikelolanya | 1 Maret 2021 |

## IAM role untuk Amazon EC2

Aplikasi harus menandatangani permintaan API mereka dengan AWS kredenSIAL. Oleh karena itu, jika Anda adalah seorang developer aplikasi, Anda membutuhkan strategi untuk mengelola kredensial untuk aplikasi-aplikasi Anda yang berjalan pada instans EC2. Sebagai contoh, Anda dapat mendistribusikan kredensial AWS Anda dengan aman ke instans, yang mana hal itu akan memungkinkan aplikasi-aplikasi pada instans tersebut untuk menggunakan kredensial Anda untuk menandatangani permintaan, sekaligus melindungi kredensial Anda dari pengguna lain. Namun, sulit untuk mendistribusikan kredentif dengan aman ke setiap instans, terutama yang dibuat AWS atas

nama Anda, seperti Instans Spot atau instans di grup Penskalaan Otomatis. Anda juga harus dapat memperbarui kredenSIAL pada setiap instance saat Anda memutar AWS kredenSIAL Anda.

Kami merancang IAM role agar aplikasi-aplikasi Anda dapat membuat permintaan API dengan aman dari instans Anda, tanpa mengharuskan Anda mengelola kredensial keamanan yang digunakan oleh aplikasi-aplikasi tersebut. Alih-alih membuat dan mendistribusikan AWS kredenSIAL Anda, Anda dapat mendelegasikan izin untuk membuat permintaan API menggunakan peran IAM sebagai berikut:

1. Buat peran IAM.
2. Tentukan akun atau AWS layanan mana yang dapat mengambil peran tersebut.
3. Tentukan tindakan dan sumber daya API mana yang dapat digunakan oleh aplikasi setelah peran tersebut diambil.
4. Tentukan peran saat Anda meluncurkan instans Anda, atau lampirkan peran tersebut ke instans yang sudah ada.
5. Buatlah aplikasi tersebut mengambil satu set kredensial sementara lalu gunakan kredensial tersebut.

Sebagai contoh, Anda dapat menggunakan IAM role untuk memberikan izin ke aplikasi-aplikasi yang berjalan pada instans Anda yang harus menggunakan bucket dalam Amazon Simple Storage Service (Amazon S3). Anda dapat menentukan izin untuk IAM role dengan membuat kebijakan dalam format JSON. Peran ini mirip dengan kebijakan yang Anda buat untuk pengguna. Jika Anda mengubah peran, maka perubahan itu akan disebarkan ke semua instans.

> **Note**
>
> KredenSIAL peran Amazon EC2 IAM tidak tunduk pada durasi sesi maksimum yang dikonfigurasi dalam peran. Untuk informasi selengkapnya, lihat Metode untuk mengambil peran dalam Panduan Pengguna IAM.

Saat Anda membuat IAM role, kaitkan kebijakan IAM dengan hak akses paling rendah yang membatasi akses ke panggilan API tertentu yang diperlukan oleh aplikasi. Untuk Windows-to-Windows komunikasi, gunakan grup dan peran Windows yang terdefinisi dengan baik dan terdokumentasi dengan baik untuk memberikan akses tingkat aplikasi antar instans Windows. Grup dan peran memungkinkan pelanggan untuk menentukan izin aplikasi dengan hak akses paling rendah dan izin tingkat folder NTFS untuk membatasi akses ke persyaratan spesifik aplikasi.

Anda hanya dapat melampirkan satu IAM role saja ke instans, tapi Anda dapat melampirkan peran yang sama ke banyak instans. Untuk informasi selengkapnya tentang cara membuat dan menggunakan IAM role, lihat Peran dalam Panduan Pengguna IAM.

Anda dapat menerapkan izin tingkat sumber daya pada kebijakan IAM Anda untuk mengontrol kemampuan pengguna dalam melampirkan, mengganti, atau melepaskan IAM role yang dilampirkan ke instans. Untuk informasi selengkapnya, lihat Izin tingkat sumber daya yang mendukung tindakan API Amazon EC2 dan contoh berikut ini: Contoh: Cara Menggunakan IAM role.

Daftar Isi


- Profil instans
- Izin untuk kasus penggunaan Anda
- Mengambil kredensial keamanan dari metadata instans
- Memberikan izin untuk melampirkan peran IAM ke instance
- Melampirkan IAM role ke instans
- Peran identitas instans untuk instans Amazon EC2


## Profil instans

Amazon EC2 menggunakan profil instans sebagai kontainer untuk IAM role. Saat Anda membuat IAM role menggunakan konsol IAM, konsol akan membuat profil instans secara otomatis dan memberikan nama yang sama sesuai dengan perannya. Jika Anda menggunakan konsol Amazon EC2 untuk meluncurkan instans dengan IAM role atau untuk melampirkan IAM role ke instans, Anda dapat memilih peran tersebut berdasarkan daftar nama profil instans.

Jika Anda menggunakan AWS CLI, API, atau AWS SDK untuk membuat peran, Anda membuat profil peran dan instance sebagai tindakan terpisah, dengan nama yang berpotensi berbeda. Jika Anda kemudian menggunakan AWS CLI, API, atau AWS SDK untuk meluncurkan instance dengan peran IAM atau untuk melampirkan peran IAM ke instance, tentukan nama profil instans.

profil instans hanya dapat berisi satu IAM role saja. Anda dapat menyertakan peran IAM di beberapa profil instans.

Untuk memperbarui izin untuk sebuah instance, ganti profil instansnya. Kami tidak menyarankan untuk menghapus peran dari profil instance, karena ada penundaan hingga satu jam sebelum perubahan ini berlaku.

Untuk informasi selengkapnya, lihat Menggunakan profil instance di Panduan Pengguna IAM.

## Izin untuk kasus penggunaan Anda

Ketika Anda pertama kali membuat IAM role untuk aplikasi-aplikasi Anda, terkadang Anda mungkin harus memberikan izin melebihi yang diperlukan. Sebelum meluncurkan aplikasi Anda di lingkungan produksi Anda, Anda dapat membuat kebijakan IAM yang berbasis aktivitas akses untuk IAM role. IAM Access Analyzer meninjau AWS CloudTrail log Anda dan membuat templat kebijakan yang berisi izin yang telah digunakan oleh peran dalam rentang tanggal yang Anda tentukan. Anda dapat menggunakan templat tersebut untuk membuat kebijakan terkelola dengan izin yang sangat terperinci dan kemudian melampirkan kebijakan itu ke IAM role. Dengan begitu, Anda hanya memberikan izin yang dibutuhkan peran untuk berinteraksi dengan AWS sumber daya untuk kasus penggunaan spesifik Anda. Hal ini akan membantu Anda untuk lebih mematuhi praktik terbaik dalam memberikan hak akses paling rendah. Untuk informasi selengkapnya, lihat pembuatan kebijakan Penganalisis Akses IAM di Panduan Pengguna IAM.

## Mengambil kredensial keamanan dari metadata instans

aplikasi pada instans akan mengambil kredensial keamanan yang disediakan oleh peran dari item metadata instans `iam/security-credentials/role-name`. Aplikasi ini diberi izin untuk tindakan-tindakan dan sumber daya yang telah Anda tentukan untuk peran tersebut melalui kredensial keamanan yang dikaitkan dengan peran tersebut. Kredensial keamanan ini bersifat sementara dan kami memutar kredensial tersebut secara otomatis. Kami menyediakan kredensial yang baru setidaknya lima menit sebelum kredensial lama kedaluwarsa.

Untuk informasi selengkapnya tentang metadata instans, lihat Menggunakan metadata instans untuk mengelola instans EC2 Anda.

> **Warning**
>
> Jika Anda menggunakan layanan-layanan yang menggunakan metadata instans dengan IAM role, pastikan Anda tidak mengekspos kredensial Anda saat layanan-layanan tersebut melakukan panggilan HTTP atas nama Anda. Jenis layanan yang dapat mengekspos kredenSIAL Anda termasuk proxy HTTP, layanan HTML/CSS validator, dan prosesor XML yang mendukung inklusi XML.

Untuk beban kerja Amazon EC2, sebaiknya Anda mengambil kredensial sesi menggunakan metode yang dijelaskan di bawah ini. Kredensial ini harus memungkinkan beban kerja Anda membuat permintaan API AWS , tanpa perlu menggunakan `sts:AssumeRole` untuk mengambil peran yang

sama yang sudah dikaitkan dengan instans. Kecuali jika Anda perlu meneruskan tanda sesi untuk kontrol akses berbasis atribut (ABAC) atau meneruskan kebijakan sesi guna membatasi izin peran lebih lanjut, panggilan pengambilan peran tersebut tidak diperlukan karena panggilan tersebut membuat kumpulan baru kredensial sesi peran sementara yang sama.

Jika beban kerja menggunakan peran untuk mengambil dirinya sendiri, Anda harus membuat kebijakan kepercayaan yang secara eksplisit memungkinkan peran tersebut untuk mengambil dirinya sendiri. Jika Anda tidak membuat kebijakan kepercayaan, Anda mendapatkan `AccessDenied` kesalahan. Untuk informasi selengkapnya, lihat Memper barui kebijakan kepercayaan peran di Panduan Pengguna IAM.

IMDSv2

Linux

Jalankan perintah berikut dari instance Linux Anda untuk mengambil kredenSIAL keamanan untuk peran IAM.

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/iam/security-credentials/role-name
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda untuk mengambil kredenSIAL keamanan untuk peran IAM.

```
[string]$token = Invoke-RestMethod `
    -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
    -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod `
    -Headers @{"X-aws-ec2-metadata-token" = $token} `
    -Method GET -Uri http://169.254.169.254/latest/meta-data/iam/security-credentials/role-name
```

IMDSv1

Linux

Jalankan perintah berikut dari instance Linux Anda untuk mengambil kredenSIAL keamanan untuk peran IAM.

```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials/role-name
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda untuk mengambil kredenSIAL keamanan untuk peran IAM.

```
Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/iam/security-
credentials/role-name
```

Berikut ini adalah output contoh. Jika Anda tidak dapat mengambil kredentif keamanan, lihat Saya tidak dapat mengakses kredenSIAL keamanan sementara pada instans EC2 saya di Panduan Pengguna IAM.

```
{
  "Code" : "Success",
  "LastUpdated" : "2012-04-26T16:39:16Z",
  "Type" : "AWS-HMAC",
  "AccessKeyId" : "ASIAIOSFODNN7EXAMPLE",
  "SecretAccessKey" : "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
  "Token" : "token",
  "Expiration" : "2017-05-17T15:09:54Z"
}
```

Untuk aplikasi AWS CLI, dan PowerShell perintah Alat untuk Windows yang berjalan pada instance, Anda tidak perlu secara eksplisit mendapatkan kredentif keamanan sementara — AWS SDK, AWS CLI, dan Alat untuk Windows PowerShell secara otomatis mendapatkan kredenSIAL dari layanan metadata instance EC2 dan menggunakannya. Untuk membuat panggilan di luar instans menggunakan kredensial keamanan sementara (sebagai contoh, untuk menguji kebijakan IAM), Anda harus menyediakan kunci akses, kunci rahasia, dan token sesi. Untuk informasi selengkapnya, lihat Menggunakan KredenSIAL Keamanan Sementara untuk Meminta Akses ke Sumber AWS Daya di Panduan Pengguna IAM.

## Memberikan izin untuk melampirkan peran IAM ke instance

Identitas Anda Akun AWS, seperti pengguna IAM, harus memiliki izin khusus untuk meluncurkan instans Amazon EC2 dengan peran IAM, melampirkan peran IAM ke instans, mengganti peran IAM untuk instans, atau melepaskan peran IAM dari instans. Anda harus memberikan izin untuk menggunakan tindakan API berikut sesuai kebutuhan:

- `iam:PassRole`
- `ec2:AssociateIamInstanceProfile`
- `ec2:DisassociateIamInstanceProfile`
- `ec2:ReplaceIamInstanceProfileAssociation`

> **Note**
>
> Jika Anda menentukan sumber daya untuk `iam:PassRole` as`*`, ini akan memberikan akses untuk meneruskan salah satu peran IAM Anda ke instance. Untuk mengikuti praktik terbaik hak istimewa ter kecil, tentukan ARN peran IAM tertentu dengan`iam:PassRole`, seperti yang ditunjukkan pada contoh kebijakan di bawah ini.

Contoh kebijakan untuk akses terprogram

Kebijakan IAM berikut memberikan izin untuk meluncurkan instans dengan peran IAM, melampirkan peran IAM ke instans, atau mengganti peran IAM untuk instance menggunakan AWS CLI atau Amazon EC2 API.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances",
         "ec2:AssociateIamInstanceProfile",
         "ec2:DisassociateIamInstanceProfile",
         "ec2:ReplaceIamInstanceProfileAssociation"
```

```
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/DevTeam*"
    }
  ]
}
```

Persyaratan tambahan untuk akses konsol

Untuk memberikan izin untuk menyelesaikan tugas yang sama menggunakan konsol Amazon EC2, Anda juga harus menyertakan tindakan `iam:ListInstanceProfiles` API.

## Melampirkan IAM role ke instans

Anda dapat membuat IAM role dan melampirkan peran itu ke instans selama atau setelah peluncuran dilakukan. Anda juga dapat mengganti atau melepaskan peran IAM.

Membuat dan melampirkan peran IAM selama peluncuran instance (Direkomendasikan)

1. Selama peluncuran instans EC2, perluas Detail lanjutan.
2. Di bagian profil instance IAM, pilih Buat peran IAM baru.
3. Formulir pembuatan peran sebaris terbuka, memungkinkan Anda untuk:

   - T entukan nama Per an (misalnya,`EC2-S3-Access-Role`)
   - Tentukan izin dengan memilih kebijakan AWS terkelola atau membuat kebijakan khusus untuk instans Anda

     Misalnya, untuk memberikan akses S3, pilih kebijakan ter `AmazonS3ReadOnlyAccess` kelola
   - Tinjau kebijakan kepercayaan yang memungkinkan `ec2.amazonaws.com` untuk mengambil peran
   - Tambahkan tag opsional untuk metadata
4. Pilih Buat peran.

   Peran yang baru dibuat dipilih secara otomatis dan akan dilampirkan ke instans Anda melalui profil instance saat instance diluncurkan.

> **Note**
>
> Saat Anda membuat peran menggunakan konsol selama peluncuran instance, profil instance dengan nama yang sama dengan peran akan dibuat secara otomatis. Profil instance adalah wadah yang meneruskan informasi peran IAM ke instance saat peluncuran.

> **Important**
>
> - Anda hanya dapat melampirkan satu IAM role saja ke instans, tapi Anda dapat melampirkan peran yang sama ke banyak instans.
> - Kaitkan kebijakan IAM hak istimewa terendah yang membatasi akses ke panggilan API tertentu yang dibutuhkan aplikasi.

Untuk informasi selengkapnya tentang cara membuat dan menggunakan IAM role, lihat Peran dalam Panduan Pengguna IAM.

Melampirkan peran IAM yang ada selama peluncuran instance

Untuk melampirkan peran IAM yang ada ke instans saat peluncuran menggunakan konsol Amazon EC2, perluas Detail lanjutan. Untuk profil instance IAM, pilih peran IAM dari daftar dropdown.

> **Note**
>
> Jika Anda membuat peran IAM menggunakan konsol IAM, profil instans dibuat untuk Anda dan diberi nama yang sama dengan peran tersebut. Jika Anda membuat peran IAM menggunakan AWS CLI, API, atau AWS SDK, Anda mungkin telah memberi profil instans nama yang berbeda dari peran tersebut.

Anda dapat melampirkan peran IAM ke instance yang sedang berjalan atau dihentikan. Jika instance sudah memiliki peran IAM yang dilampirkan, Anda harus menggantinya dengan peran IAM baru.

Console

Untuk melampirkan IAM role ke instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih T ind akan, Keamanan, Ubah peran IAM.
5. Untuk peran IAM, pilih profil instance IAM.
6. Pilih Per barui peran IAM.

AWS CLI

Untuk melampirkan IAM role ke instans

Gunakan perintah associate-iam-instance-profile untuk melampirkan peran IAM ke instance. Saat menentukan profil instance, Anda dapat menggunakan Amazon Resource Name (ARN) dari profil instans, atau Anda dapat menggunakan namanya.

```
aws ec2 associate-iam-instance-profile \
    --instance-id i-1234567890abcdef0 \
    --iam-instance-profile Name="TestRole-1"
```

PowerShell

Untuk melampirkan IAM role ke instans

Gunakan Register-EC2IamInstanceProfile cmdlet.

```
Register-EC2IamInstanceProfile `
    -InstanceId i-1234567890abcdef0 `
    -IamInstanceProfile_Name TestRole-1
```

Untuk mengganti peran IAM pada instance yang sudah memiliki peran IAM terlampir, instance harus berjalan. Anda dapat melakukan hal ini jika Anda ingin mengubah IAM role untuk instans tanpa melepaskan peran terlampir yang ada terlebih dahulu. Sebagai contoh, Anda dapat melakukan hal ini untuk memastikan bahwa tindakan-tindakan API yang dilakukan oleh aplikasi yang berjalan pada instans tidak terganggu.

Console

Untuk mengganti sebuah peran IAM ke sebuah instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih T ind akan, Keamanan, Ubah peran IAM.
5. Untuk peran IAM, pilih profil instance IAM.
6. Pilih Per barui peran IAM.

AWS CLI

Untuk mengganti IAM role untuk instans

1. Jika diperlukan, gunakan perintah describe-iam-instance-profile- association untuk mendapatkan ID asosiasi.

```
aws ec2 describe-iam-instance-profile-associations \
    --filters Name=instance-id,Values=i-1234567890abcdef0 \
    --query IamInstanceProfileAssociations.AssociationId
```

2. Gunakan perintah replace-iam-instance-profile-association. Tentukan ID asosiasi untuk profil instance yang ada dan ARN atau nama profil instance baru.

```
aws ec2 replace-iam-instance-profile-association \
    --association-id iip-assoc-0044d817db6c0a4ba \
    --iam-instance-profile Name="TestRole-2"
```

PowerShell

Untuk mengganti IAM role untuk instans

1. Jika diperlukan, gunakan Get-EC2IamInstanceProfileAssociation cmdlet untuk mendapatkan ID asosiasi.

```
(Get-EC2IamInstanceProfileAssociation -Filter @{Name="instance-id";
 Values="i-0636508011d8e966a"}).AssociationId
```

2. Gunakan Set-EC2IamInstanceProfileAssociation cmdlet. Tentukan ID asosiasi untuk profil instance yang ada dan ARN atau nama profil instance baru.

```
Set-EC2IamInstanceProfileAssociation `
```

```
    -AssociationId iip-assoc-0044d817db6c0a4ba `
    -IamInstanceProfile_Name TestRole-2
```

Anda dapat melepaskan peran IAM dari instance yang sedang berjalan atau dihentikan.

Console

Untuk melepaskan IAM role terlampir dari instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih T ind akan, Keamanan, Ubah peran IAM.
5. Untuk IAM role, pilih No IAM Role (Tanpa IAM Role).
6. Pilih Per barui peran IAM.
7. Saat dipromosikan untuk konfirmasi, masukkan Lepaskan, lalu pilih Lepaskan.

AWS CLI

Untuk melepaskan IAM role terlampir dari instans

1. Jika diperlukan, gunakan describe-iam-instance-profile-association untuk mendapatkan ID asosiasi untuk profil instance IAM untuk dilepaskan.

```
aws ec2 describe-iam-instance-profile-associations \
    --filters Name=instance-id,Values=i-1234567890abcdef0 \
    --query IamInstanceProfileAssociations.AssociationId
```

2. Gunakan perintah disassociate- iam-instance-profile.

```
aws ec2 disassociate-iam-instance-profile --association-id iip-
assoc-0044d817db6c0a4ba
```

PowerShell

Untuk melepaskan IAM role terlampir dari instans

1. Jika diperlukan, gunakan Get-EC2IamInstanceProfileAssociation untuk mendapatkan ID asosiasi untuk profil instance IAM untuk dilepaskan.

```
(Get-EC2IamInstanceProfileAssociation -Filter @{Name="instance-id";
 Values="i-0636508011d8e966a"}).AssociationId
```

2. Gunakan Unregister-EC2IamInstanceProfile cmdlet.

```
Unregister-EC2IamInstanceProfile -AssociationId iip-assoc-0044d817db6c0a4ba
```

## Peran identitas instans untuk instans Amazon EC2

Setiap instans Amazon EC2 yang Anda luncurkan memiliki peran identitas instans yang mewakili identitasnya. Peran identitas instance adalah jenis peran IAM. AWS layanan dan fitur yang terintegrasi untuk menggunakan peran identitas instance dapat menggunakannya untuk mengidentifikasi instance ke layanan.

Kredensional peran identitas instans dapat diakses dari Layanan Metadata Instans (IMDS) di `/identity-credentials/ec2/security-credentials/ec2-instance`. KredenSIAL terdiri dari pasangan kunci akses AWS sementara dan token sesi. Mereka digunakan untuk menandatangani permintaan AWS Sigv4 ke AWS layanan yang menggunakan peran identitas instance. Kredensial hadir dalam metadata instans terlepas dari apakah layanan atau fitur yang menggunakan peran identitas instans diaktifkan pada instans.

Peran identitas instans dibuat secara otomatis saat instance diluncurkan, tidak memiliki dokumen kebijakan role-trust, dan tidak tunduk pada identitas atau kebijakan sumber daya apa pun.

Layanan yang didukung

AWS Layanan berikut menggunakan peran identitas instance:

- Amazon Bedrock AgentCore — Instans AgentCore Runtime menggunakan peran identitas instans untuk menyiapkan instans selama peluncuran.
- Amazon EC2 — EC2 Instance Connect menggunakan peran identitas instans untuk memperbarui kunci host untuk instans Linux.

- Amazon GuardDuty — GuardDuty Runtime Monit oring menggunakan peran identitas instans untuk mengizinkan agen runtime mengirim telemetri keamanan ke titik akhir GuardDuty VPC.
- AWS Lambda— Instans Terkelola Lambda menggunakan peran identitas instance untuk kait siklus hidup, telemetri, dan distribusi artefak.
- AWS Security Token Service (AWS STS) — Kredentif peran identitas instans dapat digunakan dengan AWS STS `GetCallerIdentity` tindakan.
- AWS Systems Manager- Saat menggunakan Konfigur asi Manajemen Host Default, AWS Systems Manager menggunakan identitas yang disediakan oleh peran identitas instans untuk mendaftarkan instans EC2. Setelah mengidentifikasi instans Anda, Systems Manager dapat meneruskan peran IAM `AWSSystemsManagerDefaultEC2InstanceManagementRole` ke instans Anda.

Peran identitas instans tidak dapat digunakan dengan AWS layanan atau fitur lain karena tidak memiliki integrasi dengan peran identitas instance.

ARN peran identitas instans

ARN peran identitas instans ARN mengambil format berikut:

```
arn:aws-partition:iam::account-number:assumed-role/aws:ec2-instance/instance-id
```

Misalnya:

```
arn:aws:iam::0123456789012:assumed-role/aws:ec2-instance/i-1234567890abcdef0
```

Untuk informasi selengkapnya tentang ARN, lihat Amazon Resource Names (ARN) di Panduan Pengguna IAM.

## Manajemen pembaruan untuk instans Amazon EC2

Kami merekomendasikan agar Anda melakukan patch, memperbarui, dan mengamankan sistem operasi dan aplikasi pada instans EC2 Anda secara berkala. Anda dapat menggunakan AWS Systems Manager Patch Manager untuk mengotomatisasi proses penginstalan pembaruan terkait keamanan untuk sistem operasi maupun aplikasi.

Untuk instans EC2 dalam grup Auto Scaling, Anda dapat menggunakan runbook AWS-PatchAsgInstance untuk membantu menghindari penggantian instans yang sedang di-patch. Atau, Anda dapat menggunakan layanan pembaruan otomatis atau proses yang direkomendasikan untuk menginstal pembaruan yang disediakan oleh vendor aplikasi.

Sumber daya

- AL2023 — Memper barui AL2023 di Panduan Pengguna Amazon Linux 2023
- AL2 — Kelola perangkat lunak pada instans Amazon Linux 2 Anda di Panduan Pengguna Amazon Linux 2
- Instans Windows — the section called “Manajemen pembaruan”

# Praktik terbaik keamanan untuk instans Windows

Sebaiknya ikuti praktik terbaik keamanan ini untuk instans Windows Anda.

Daftar Isi


- High-level praktik terbaik keamanan
- Manajemen pembaruan
- Manajemen konfigurasi
- Manajemen perubahan
- Audit dan akuntabilitas untuk instans Windows Amazon EC2


## High-level praktik terbaik keamanan

Anda harus mematuhi praktik terbaik keamanan tingkat tinggi berikut untuk instans Windows Anda:

- Akses terkecil — Berikan akses hanya ke sistem dan lokasi yang tepercaya dan diharapkan. Hal ini berlaku untuk semua produk Microsoft, seperti Active Directory, server produktivitas bisnis Microsoft, serta layanan infrastruktur, seperti Remote Desktop Services, server proksi terbalik, server web IIS, dan lainnya. Gunakan AWS kemampuan seperti grup keamanan instans Amazon EC2, daftar kontrol akses jaringan (ACL), dan public/private subnet Amazon VPC untuk melapisi keamanan di beberapa lokasi dalam arsitektur. Dalam instans Windows, para pelanggan dapat menggunakan Windows Firewall untuk melapisi lebih jauh strategi pertahanan yang mendalam dalam deployment mereka. Cukup instal komponen dan aplikasi OS yang diperlukan agar sistem berfungsi sebagaimana peruntukannya. Konfigurasikan layanan infrastruktur, seperti IIS, untuk dijalankan di bawah akun layanan atau untuk menggunakan fitur, seperti identitas kolam aplikasi, agar dapat mengakses sumber daya secara lokal dan jarak jauh di seluruh infrastruktur Anda.
- Hak istimewa terkecil — Tentukan kumpulan hak istimewa minimum yang dibutuhkan instans dan akun untuk menjalankan fungsinya. Batasi server dan pengguna tersebut agar hanya

memperbolehkan izin yang ditentukan ini. Gunakan teknik, seperti Kontrol Akses Berbasis Peran, untuk mengurangi luas permukaan akun administratif dan membuat peran paling terbatas untuk menyelesaikan tugas. Gunakan fitur OS, seperti Encrypting File System (EFS), di dalam NTFS untuk mengenkripsi data diam yang sensitif serta mengontrol akses aplikasi dan pengguna ke data diam tersebut.

- Manajemen konfigurasi — Buat konfigurasi server dasar yang menggabungkan patch keamanan terkini dan suite perlindungan berbasis host yang mencakup anti-virus, anti-malware, intrusi detection/prevention, dan pemantauan integritas file. Nilai setiap server menurut data baseline yang tercatat saat ini untuk mengidentifikasi dan menandai setiap deviasi. Pastikan setiap server dikonfigurasi untuk menghasilkan serta menyimpan data log dan audit yang sesuai dengan aman.
- Manajemen perubahan — Buat proses untuk mengontrol perubahan pada garis dasar konfigurasi server dan bekerja menuju proses perubahan yang sepenuhnya otomatis. Manfaatkan juga Just Enough Administration (JEA) dengan Windows PowerShell DSC untun membatasi akses administratif ke fungsi minimum yang diperlukan.
- Manajemen patch — Terapkan proses yang secara teratur menambal, memperbarui, dan mengamankan sistem operasi dan aplikasi pada instans EC2 Anda.
- Log audit — Mengaudit akses dan semua perubahan pada instans Amazon EC2 untuk memverifikasi integritas server dan memastikan hanya perubahan resmi yang dilakukan. Manfaatkan fitur seperti Enhanced Logging for IIS untuk meningkatkan kemampuan pencatatan default. AWS kemampuan seperti Log Aliran VPC dan juga AWS CloudTrail tersedia untuk mengaudit akses jaringan, termasuk allowed/denied permintaan dan panggilan API, masing-masing.

## Manajemen pembaruan

Untuk memastikan hasil terbaik saat menjalankan Windows Server di Amazon EC2, kami sarankan Anda menerapkan praktik terbaik berikut:

- Configure Windows Update
- Update drivers
- Use the latest Windows AMIs
- Test performance before migration
- Update launch agents
- Mulai ulang instans Windows Anda setelah Anda menginstal pembaruan. Untuk informasi selengkapnya, lihat Mulai ulang instans Amazon EC2 Anda.

Untuk informasi selengkapnya, tentang cara meningkatkan atau migrasi instans Windows ke versi Windows Server yang lebih baru, lihat Tingkatkan instance EC2 Windows ke versi Windows Server yang lebih baru.

Konfigurasikan Pembaruan Windows

Secara default, instans yang diluncurkan dari AMI AWS Windows Server tidak menerima pembaruan melalui Pembaruan Windows.

Perbarui driver Windows

Pertahankan driver terbaru di semua instans Windows EC2 untuk memastikan bahwa perbaikan masalah dan peningkatan performa terbaru telah diterapkan di seluruh armada Anda. Tergantung pada jenis instans Anda, Anda harus memperbarui driver AWS PV, Amazon ENA, dan AWS NVMe.

- Gunakan topik SNS untuk menerima pembaruan untuk rilis driver baru.
- Gunakan runbook AWS Systems Manager Otomasi AWSSupport-UpgradeWindowsAWSDrivers untuk menerapkan pembaruan dengan mudah di seluruh instans Anda.

Luncurkan instance menggunakan AMI Windows terbaru

AWS merilis AMI Windows baru setiap bulan, yang berisi patch OS terbaru, driver, dan agen peluncuran. Anda harus memanfaatkan AMI terkini saat meluncurkan instans baru atau saat membangun gambar kustom Anda sendiri.

- Untuk melihat pembaruan untuk setiap rilis AMI AWS Windows, lihat ri AWS wayat versi AMI Windows.
- Untuk membangun dengan AMI terkini yang tersedia, lihat Kueri untuk AMI Windows Terkini Menggunakan Systems Manager Parameter Store.
- Untuk informasi selengkapnya tentang AMI Windows khusus yang dapat Anda gunakan untuk meluncurkan instans untuk database dan kasus penggunaan pengerasan kepatuhan, lihat AMI Windows Khus us di Refer AWS ensi AMI Windows.

Uji system/application kinerja sebelum migrasi

Migrasi aplikasi perusahaan untuk AWS dapat melibatkan banyak variabel dan konfigurasi. Selalu uji performa solusi EC2 untuk memastikan bahwa:

- Tipe Instans dikonfigurasi dengan benar, termasuk ukuran instans, peningkatan jaringan, dan penghunian (bersama atau khusus).
- Topologi instans sesuai untuk beban kerja dan memanfaatkan fitur berperforma tinggi bila diperlukan, seperti penghunian khusus, grup penempatan, volume penyimpanan instans, dan bare metal.

Instal EC2Launch v2 versi terbaru

Instal agen EC2launch v2 terbaru untuk memastikan bahwa peningkatan terbaru diterapkan di seluruh armada Anda. Untuk informasi selengkapnya, lihat Instal EC2launch v2.

Jika Anda memiliki armada campuran, atau jika ingin terus menggunakan agen EC2Launch (Windows Server 2016 dan 2019) atau EC2 Config (khusus OS warisan), perbarui ke versi terbaru agen masing-masing.

Pembaruan otomatis didukung pada kombinasi versi Windows Server dan agen peluncuran berikut. Anda dapat memilih pembaruan otomatis di konsol Manajemen Host Pengaturan Cepat SSM di bawah Agen Peluncuran Amazon EC2.

| Versi Windows | EC2Launch v1 | EC2Launch v2 |
|---|---|---|
| 2016 | ✓ | ✓ |
| 2019 | ✓ | ✓ |
| 2022 | | ✓ |

- Untuk informasi selengkapnya tentang memperbarui ke EC2launch v2, lihat. the section called “Instal EC2launch v2”
- Untuk informasi untuk memperbarui EC2config secara manual, lihat. the section called “Instal EC2Config”
- Untuk informasi untuk memperbarui EC2launch secara manual, lihat. the section called “Instal EC2Launch”

# Manajemen konfigurasi

Amazon Machine Image (AMI) menyediakan konfigurasi awal untuk instans Amazon EC2 yang mencakup OS Windows dan kustomisasi spesifik pelanggan yang bersifat opsional, seperti kontrol aplikasi dan keamanan. Buat katalog AMI yang berisi garis dasar konfigurasi keamanan yang disesuaikan untuk memastikan bahwa semua instans Windows diluncurkan dengan kontrol keamanan standar. Garis dasar keamanan dapat dimasukkan ke dalam AMI, di-bootstrap secara dinamis saat instance EC2 diluncurkan, atau dikemas sebagai produk untuk distribusi seragam melalui portofolio Katalog Layanan. AWS Untuk informasi selengkapnya tentang cara mengamankan AMI, lihat Praktik Terbaik untuk Membangun AMI.

Setiap instans Amazon EC2 harus mematuhi standar keamanan organisasi. Jangan menginstal peran dan fitur Windows apa pun yang tidak diperlukan, dan instal perangkat lunak sebagai perlindungan terhadap kode berbahaya (mitigasi antivirus, antimalware, eksploitasi), pantau integritas host, dan lakukan deteksi intrusi. Lakukan konfigurasi pada perangkat lunak keamanan untuk memantau dan mempertahankan pengaturan keamanan OS, melindungi integritas file OS penting, dan mewaspadai penyimpangan dari garis dasar keamanan. Pertimbangkan untuk melaksanakan rekomendasi tolok ukur konfigurasi keamanan yang diterbitkan oleh Microsoft, Center for Internet Security (CIS), atau National Institute of Standards and Technology (NIST). Pertimbangkan untuk menggunakan alat-alat Microsoft lainnya untuk server aplikasi tertentu, seperti Penganalisis Praktik Terbaik untuk SQL Server.

AWS pelanggan juga dapat menjalankan penilaian Amazon Inspector untuk meningkatkan keamanan dan kepatuhan aplikasi yang digunakan pada instans Amazon EC2. Amazon Inspector secara otomatis menilai aplikasi dalam hal kelemahan atau penyimpangannya dari praktik terbaik dan menyertakan basis pengetahuan dari ratusan aturan yang dipetakan ke standar kepatuhan keamanan umum (misalnya PCI DSS) dan definisi kelemahan. Contoh-contoh aturan bawaan termasuk pemeriksaan apakah cara masuk dari root jarak jauh diaktifkan, atau apakah ada versi perangkat lunak yang lemah yang sudah diinstal. Aturan-aturan ini diperbarui secara berkala oleh peneliti AWS keamanan.

Saat mengamankan instans Windows, kami menyarankan Anda untuk menerapkan Layanan Domain Direktori Aktif agar dapat mengaktifkan infrastruktur yang dapat diskalakan, aman, dan dapat dikelola untuk lokasi yang didistribusikan. Selain itu, setelah meluncurkan instans dari konsol Amazon EC2 atau dengan menggunakan alat penyediaan Amazon EC2, seperti AWS CloudFormation, adalah praktik yang baik untuk menggunakan fitur OS asli, seperti Microsoft Windows PowerShell DSC untuk mempertahankan status konfigurasi jika terjadi penyimpangan konfigurasi.

## Manajemen perubahan

Setelah garis dasar keamanan awal diterapkan ke instans Amazon EC2 pada saat peluncuran, kendalikan perubahan Amazon EC2 yang terjadi untuk mempertahankan keamanan mesin virtual Anda. Menetapkan proses manajemen perubahan untuk mengotorisasi dan menggabungkan perubahan pada AWS sumber daya (seperti grup keamanan, tabel rute, dan ACL jaringan) serta konfigurasi OS dan aplikasi (seperti Windows atau tambalan aplikasi, peningkatan perangkat lunak, atau pembaruan file konfigurasi).

AWS menyediakan beberapa alat untuk membantu mengelola perubahan pada AWS sumber daya, termasuk AWS CloudTrail, AWS Config, CloudFormation, dan AWS Elastic Beanstalk, dan paket manajemen untuk System Center Operations Manager dan System Center Virtual Machine Manager. Perhatikan bahwa Microsoft merilis patch Windows pada hari Selasa kedua setiap bulan (atau sesuai kebutuhan) dan AWS memperbarui semua AMI Windows yang dikelola AWS dalam waktu lima hari setelah Microsoft merilis patch. Oleh karena itu, penting untuk terus menambal semua AMI dasar, memperbarui CloudFormation template dan konfigurasi grup Penskalaan Otomatis dengan ID AMI terbaru, dan menerapkan alat untuk mengotomatiskan manajemen patch instans yang sedang berjalan.

Microsoft menyediakan beberapa opsi untuk mengelola perubahan OS Windows dan aplikasi. SCCM, contohnya, menyediakan cakupan siklus hidup penuh modifikasi lingkungan. Pilihlah alat-alat yang memenuhi persyaratan bisnis dan lakukan kontrol pada bagaimana perubahan akan memengaruhi prosedur SLA aplikasi, kapasitas, prosedur keamanan, dan pemulihan bencana. Hindari perubahan manual dan sebagai gantinya manfaatkan perangkat lunak manajemen konfigurasi otomatis atau alat baris perintah seperti EC2 Run Command atau Windows PowerShell untuk mengimplementasikan proses perubahan berulang yang ditulis dengan skrip. Untuk membantu memenuhi persyaratan ini, gunakan host bastion dengan peningkatan pencatatan untuk semua interaksi dengan instans Windows Anda untuk memastikan bahwa semua peristiwa dan tugas direkam secara otomatis.

## Audit dan akuntabilitas untuk instans Windows Amazon EC2

AWS CloudTrail, AWS Config, dan Aturan AWS Config menyediakan fitur pelacakan audit dan perubahan untuk mengaudit perubahan AWS sumber daya. Lakukan konfigurasi pada log peristiwa Windows untuk mengirimkan file log lokal ke sistem manajemen log terpusat untuk menyimpan data log untuk digunakan dalam analisis perilaku keamanan dan operasional. Microsoft System Center Operations Manager (SCOM) mengumpulkan informasi mengenai aplikasi Microsoft yang di-deploy ke instans Windows dan menerapkan serangkaian aturan yang telah dikonfigurasi sebelumnya dan serangkaian aturan kustom berdasarkan peran dan layanan aplikasi. System Center Management

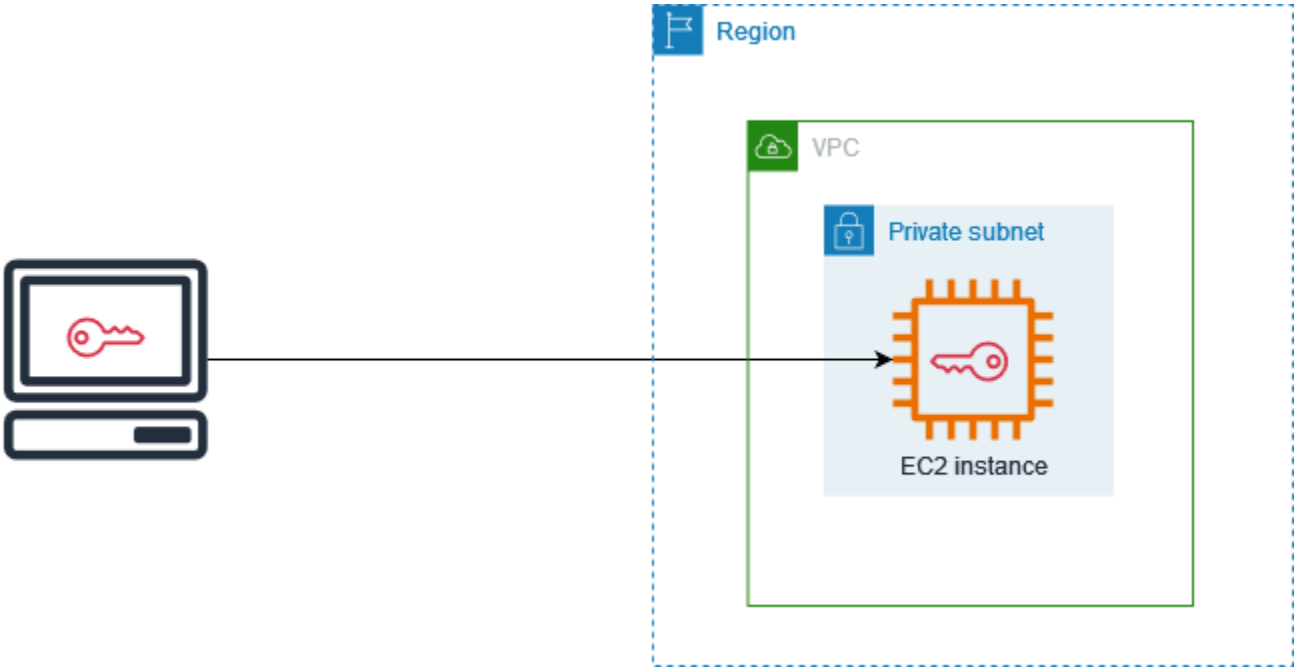
Packs yang dibangun di atas SCOM akan menyediakan pemantauan dan panduan konfigurasi spesifik aplikasi. Paket Manajemen ini mendukung Windows Server Active Directory, SharePoint Server 2013, Exchange Server 2013, Lync Server 2013, SQL Server 2014, dan banyak lagi server dan teknologi.

Selain alat manajemen sistem Microsoft, pelanggan dapat menggunakan Amazon CloudWatch untuk memantau pemanfaatan CPU instans, kinerja disk, jaringan I/O, dan melakukan pemeriksaan status host dan instans. Agen peluncuran EC2config, EC2launch, dan EC2launch v2 menyediakan akses ke fitur lanjutan tambahan untuk instans Windows. Misalnya, mereka dapat mengekspor log sistem Windows, keamanan, aplikasi, dan Layanan Informasi Internet (IIS) ke CloudWatch Log yang kemudian dapat diintegrasikan dengan CloudWatch metrik dan alarm Amazon. Pelanggan juga dapat membuat skrip yang mengekspor penghitung kinerja Windows ke metrik CloudWatch khusus Amazon.

# Pasangan kunci Amazon EC2 dan instans Amazon EC2

pasangan kunci, yang terdiri dari kunci publik dan kunci privat, adalah satu set kredensial keamanan yang Anda gunakan untuk membuktikan identitas Anda ketika terhubung ke instans Amazon EC2. Untuk instance Linux, kunci pribadi memungkinkan Anda untuk SSH dengan aman ke dalam instance Anda. Untuk instans Windows, kunci pribadi diperlukan untuk mendekripsi kata sandi administrator, yang kemudian Anda gunakan untuk menyambung ke instance Anda.

Amazon EC2 menyimpan kunci publik pada instans Anda, dan Anda menyimpan kunci pribadi, seperti yang ditunjukkan pada diagram berikut. Penting bagi Anda untuk menyimpan kunci pribadi Anda di tempat yang aman karena siapa pun yang memiliki kunci pribadi Anda dapat terhubung ke instans Anda yang menggunakan pasangan kunci.




Saat meluncurkan instance, Anda dapat menentukan pasangan kunci, sehingga Anda dapat terhubung ke instans menggunakan metode yang memerlukan pasangan kunci. Bergantung pada cara Anda mengelola keamanan, Anda dapat menentukan pasangan kunci yang sama untuk semua instans atau Anda dapat menentukan pasangan kunci yang berbeda.

Untuk instance Linux, saat instance Anda melakukan booting untuk pertama kalinya, kunci publik yang Anda tentukan saat peluncuran ditempatkan pada instance Linux Anda dalam entri di dalam`~/.ssh/authorized_keys`. Ketika Anda terhubung ke instans Linux Anda menggunakan SSH, untuk log in Anda harus menentukan kunci privat yang sesuai dengan kunci publik.

Untuk informasi selengkapnya tentang menghubungkan ke instans EC2 Anda, lihatSambungkan ke instans EC2.

**Important**

Karena Amazon EC2 tidak menyimpan salinan kunci privat Anda, tidak ada cara yang bisa dilakukan untuk memulihkan kunci privat tersebut jika Anda menghilangkannya. Akan tetapi, masih ada cara untuk terhubung ke instans yang kunci privatnya hilang. Untuk informasi selengkapnya, lihat Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance saya?

Sebagai alternatif untuk pasangan kunci, Anda dapat menggunakan AWS Systems Manager Session Manager untuk terhubung ke instance Anda dengan shell berbasis browser interaktif atau AWS Command Line Interface (AWS CLI).

Daftar Isi


- Membuat pasangan kunci untuk instans Amazon EC2
- Jelaskan pasangan kunci Anda
- Menghapus pasangan kunci Anda
- Menambahkan atau mengganti kunci publik pada instans Linux Anda
- Lakukan verifikasi terhadap sidik jari pasangan kunci Anda


# Membuat pasangan kunci untuk instans Amazon EC2

Anda dapat menggunakan Amazon EC2 untuk membuat pasangan kunci, atau Anda dapat menggunakan alat pihak ketiga untuk membuat pasangan kunci Anda, lalu mengimpornya ke Amazon EC2.

Amazon EC2 mendukung kunci SSH-2 RSA 2048-bit untuk instans Linux dan Windows. Amazon EC2 juga mendukung kunci ED25519 untuk instans Linux.

Untuk petunjuk tentang cara menyambung ke instance setelah Anda membuat pasangan kunci, lihat the section called “Hubungkan ke instans Linux Anda, menggunakan SSH.” danthe section called “Hubungkan ke instans Windows Anda menggunakan RDP”.

Daftar Isi


- Membuat pasangan kunci menggunakan Amazon EC2
- Buat pasangan kunci menggunakan AWS CloudFormation
- Membuat pasangan kunci menggunakan alat pihak ketiga dan mengimpor kunci publik ke Amazon EC2


## Membuat pasangan kunci menggunakan Amazon EC2

Saat Anda membuat pasangan kunci menggunakan Amazon EC2, kunci publik akan disimpan di Amazon EC2, dan Anda akan menyimpan kunci privat.

Anda dapat membuat hingga 5.000 pasangan kunci per Wilayah. Untuk meminta peningkatan, buat kasus dukungan. Untuk informasi selengkapnya, lihat Membuat kasus dukungan di Panduan Dukungan Pengguna.

Console

Untuk membuat pasangan kunci menggunakan Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, di Jaringan & Keamanan, pilih Pasangan Kunci.
3. Pilih Buat pasangan kunci.
4. Untuk Nama, masukkan nama deskriptif untuk pasangan kunci tersebut. Amazon EC2 akan mengaitkan kunci publik dengan nama yang Anda cantumkan sebagai nama kunci. Nama kunci dapat terdiri dari hingga 255 karakter ASCII. Tidak boleh mengandung spasi di depan maupun belakang.
5. Pilih jenis pasangan kunci yang sesuai untuk sistem operasi Anda:

   (Instans Linux) Untuk tipe pasangan kunci, pilih RSA atau ED 25519.

   (Instans Windows) Untuk jenis pasangan kunci, pilih RSA. Kunci ED25519 tidak didukung untuk instans Windows.
6. Untuk Format file kunci privat, pilih format untuk menyimpan kunci privat tersebut. Untuk menyimpan kunci privat dalam format yang dapat digunakan dengan OpenSSH, pilih pem. Untuk menyimpan kunci privat dalam format yang dapat digunakan dengan PuTTY, pilih ppk.
7. Untuk menambahkan tanda ke kunci publik, pilih Tambah tanda (Tambahkan tanda), lalu masukkan kunci dan nilai untuk tanda tersebut. Ulangi hal itu untuk setiap tanda.
8. Pilih Buat pasangan kunci.
9. File kunci privat tersebut akan secara otomatis diunduh oleh peramban Anda. Nama file dasar adalah nama yang Anda tentukan sebagai nama pasangan kunci Anda, dan ekstensi dari nama file tersebut ditentukan oleh format file yang Anda pilih. Simpan file kunci privat di suatu tempat yang aman.

   > **Important**
   >
   > Ini adalah satu-satunya kesempatan Anda untuk menyimpan file kunci privat tersebut.

10. Jika Anda berencana menggunakan klien SSH pada komputer macOS atau Linux untuk terhubung ke instans Linux, gunakan perintah berikut untuk mengatur izin file kunci privat Anda sehingga hanya Anda yang dapat membacanya.

```
chmod 400 key-pair-name.pem
```

Jika Anda tidak mengatur izin tersebut, Anda tidak akan dapat terhubung ke instans Anda menggunakan pasangan kunci ini. Untuk informasi selengkapnya, lihat Kesalahan: File kunci privat yang tidak dilindungi.

AWS CLI

Untuk membuat pasangan kunci menggunakan Amazon EC2

1. Gunakan perintah create-key-pair seperti berikut untuk membuat pasangan kunci dan untuk menyimpan kunci privat dalam file `.pem`. `--query`Opsi mencetak materi kunci pribadi ke output. `--output`Opsi menyimpan materi kunci pribadi dalam file yang ditentukan. Ekstensi harus berupa `.pem` salah satu atau`.ppk`, tergantung pada format kunci. Nama kunci pribadi dapat berbeda dari nama kunci publik, tetapi untuk kemudahan penggunaan, gunakan nama yang sama.

```
aws ec2 create-key-pair \
    --key-name my-key-pair \
    --key-type rsa \
    --key-format pem \
    --query "KeyMaterial" \
    --output text > my-key-pair.pem
```

2. Jika Anda berencana menggunakan klien SSH pada komputer macOS atau Linux untuk terhubung ke instans Linux, gunakan perintah berikut untuk mengatur izin file kunci privat Anda sehingga hanya Anda yang dapat membacanya.

```
chmod 400 key-pair-name.pem
```

Jika Anda tidak mengatur izin tersebut, Anda tidak akan dapat terhubung ke instans Anda menggunakan pasangan kunci ini. Untuk informasi selengkapnya, lihat Kesalahan: File kunci privat yang tidak dilindungi.

PowerShell

Untuk membuat pasangan kunci menggunakan Amazon EC2

Gunakan New-EC2KeyPair cmdlet sebagai berikut untuk menghasilkan kunci dan menyimpannya ke file `.pem` atau `.ppk`. Out-FileCmdlet menyimpan materi kunci pribadi dalam file dengan ekstensi yang ditentukan. Ekstensi harus berupa `.pem` salah satu atau `.ppk`, tergantung pada format kunci. Nama kunci pribadi dapat berbeda dari nama kunci publik, tetapi untuk kemudahan penggunaan, gunakan nama yang sama.

```
(New-EC2KeyPair `
    -KeyName "my-key-pair" `
    -KeyType "rsa" `
    -KeyFormat "pem").KeyMaterial | Out-File -Encoding ascii -FilePath C:\path\my-key-pair.pem
```

## Buat pasangan kunci menggunakan AWS CloudFormation

Saat Anda membuat pasangan kunci baru menggunakan CloudFormation, kunci pribadi disimpan ke Penyimpanan AWS Systems Manager Parameter. Nama parameter memiliki format berikut:

```
/ec2/keypair/key_pair_id
```

Untuk informasi selengkapnya, lihat Penyimpanan Parameter AWS Systems Manager dalam Panduan Pengguna AWS Systems Manager .

Untuk membuat pasangan kunci menggunakan CloudFormation

1. Tentukan KeyPair sumber daya AWS: :EC2:: di template Anda.

```
Resources:
  NewKeyPair:
    Type: 'AWS::EC2::KeyPair'
    Properties:
      KeyName: new-key-pair
```

2. Gunakan perintah describe-key-pairs sebagai berikut untuk mendapatkan ID dari pasangan kunci.

```
aws ec2 describe-key-pairs --filters Name=key-name,Values=new-key-pair --query
 KeyPairs[*].KeyPairId --output text
```

Berikut ini adalah output contoh.

```
key-05abb699beEXAMPLE
```

3. Gunakan perintah get-parameter sebagai berikut untuk mendapatkan parameter kunci Anda dan menyimpan materi kunci dalam file `.pem`.

```
aws ssm get-parameter --name /ec2/keypair/key-05abb699beEXAMPLE --with-decryption
 --query Parameter.Value --output text > new-key-pair.pem
```

Izin IAM yang diperlukan

CloudFormation Untuk mengaktifkan pengelolaan parameter Penyimpanan Parameter atas nama Anda, peran IAM yang diasumsikan oleh CloudFormation atau pengguna Anda harus memiliki izin berikut:

- `ssm:PutParameter` – Memberikan izin guna membuat parameter untuk materi kunci privat.
- `ssm:DeleteParameter` – Memberikan izin guna menghapus parameter yang menyimpan materi kunci privat. Izin ini diperlukan apakah pasangan kunci diimpor atau dibuat oleh CloudFormation.

Saat CloudFormation menghapus pasangan kunci yang dibuat atau diimpor oleh tumpukan, ia melakukan pemeriksaan izin untuk menentukan apakah Anda memiliki izin untuk menghapus parameter, CloudFormation meskipun membuat parameter hanya saat membuat pasangan kunci, bukan saat mengimpor pasangan kunci. CloudFormation menguji izin yang diperlukan menggunakan nama parameter buatan yang tidak cocok dengan parameter apa pun di akun Anda. Oleh karena itu, Anda mungkin melihat nama parameter fabrikasi dalam pesan kesalahan `AccessDeniedException`.

## Membuat pasangan kunci menggunakan alat pihak ketiga dan mengimpor kunci publik ke Amazon EC2

Alih-alih menggunakan Amazon EC2 untuk membuat pasangan kunci, Anda dapat membuat pasangan kunci RSA atau ED25519 dengan menggunakan alat pihak ketiga dan kemudian mengimpor kunci publik ke Amazon EC2.

Ketentuan untuk pasangan kunci

- Jenis yang didukung:
  - (Linux dan Windows) RSA
  - (Hanya Linux) ED25519

    > **Note**
    >
    > Kunci ED25519 tidak didukung untuk instans Windows.

  - Amazon EC2 tidak menerima kunci DSA.
- Format yang didukung:
  - Format kunci publik OpenSSH (untuk Linux, format di`~/.ssh/authorized_keys`)
  - (Hanya Linux) Jika Anda terhubung menggunakan SSH saat menggunakan API Connect Instance EC2, format SSH2 juga didukung.
  - Format file kunci privat SSH harus dalam format PEM atau PPK
  - (Khusus RSA) Format DER dengan enkode Base64
  - (Khusus RSA) Format file kunci publik SSH sebagaimana yang ditentukan dalam RFC 4716
- Panjang yang didukung:
  - 1024, 2048, dan 4096.
  - (Hanya Linux) Jika Anda terhubung menggunakan SSH saat menggunakan API Connect Instance EC2, panjang yang didukung adalah 2048 dan 4096.

Cara membuat pasangan kunci menggunakan alat pihak ketiga

1. Buat pasangan kunci dengan alat pihak ketiga yang Anda kehendaki. Sebagai contoh, Anda dapat menggunakan ssh-keygen (alat yang disediakan bersamaan dengan instalasi OpenSSH standar). Atau, Java, Ruby, Python, dan banyak bahasa pemrograman lainnya menyediakan pustaka standar yang dapat Anda gunakan untuk membuat pasangan kunci.

   > **Important**
   >
   > Kunci privat harus berupa format PEM atau PPK. Sebagai contoh, gunakan `ssh-keygen -m PEM` untuk membuat kunci OpenSSH dalam format PEM.

2. Simpan kunci publik ke file lokal. Misalnya, `~/.ssh/my-key-pair.pub` (Linux, macOS) atau `C:\keys\my-key-pair.pub` (Windows). Ekstensi nama file untuk file ini bukan hal penting.
3. Simpan kunci privat ke file lokal yang memiliki ekstensi `.pem` atau `.ppk`. Misalnya, `~/.ssh/my-key-pair.pem` atau `~/.ssh/my-key-pair.ppk` (Linux, macOS) atau `C:\keys\my-key-pair.pem` atau `C:\keys\my-key-pair.ppk` (Windows). Ekstensi file penting karena, tergantung pada alat yang Anda gunakan untuk terhubung ke instance Anda, Anda memerlukan format file tertentu. OpenSSH membutuhkan `.pem` file, sedangkan PuTTY membutuhkan `.ppk` file.

> **Important**
>
> Simpan file kunci privat di suatu tempat yang aman. Anda harus memberikan nama kunci publik Anda saat meluncurkan instans, dan nama kunci privat yang terkait setiap kali Anda terhubung dengan instans tersebut.

Setelah Anda membuat pasangan kunci, gunakan salah satu metode berikut ini untuk mengimpor kunci publik Anda ke Amazon EC2.

Console

Cara mengimpor kunci publik ke Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Pasangan Kunci.
3. Pilih Impor pasangan kunci.
4. Untuk Nama, masukkan nama deskriptif untuk kunci publik. Nama dapat terdiri dari hingga 255 karakter ASCII. Tidak termasuk spasi di bagian paling depan dan paling belakang.

> **Note**
>
> Ketika Anda terhubung ke instans Anda dari konsol EC2, konsol akan menyarankan nama ini untuk nama file kunci privat Anda.

5. Pilih Browse (Jelajah) untuk melakukan navigasi ke dan memilih kunci publik Anda, atau tempelkan konten kunci publik Anda ke bidang Konten kunci publik.
6. Pilih Impor pasangan kunci.

7. Pastikan kunci publik yang Anda impor muncul dalam daftar pasangan kunci.

AWS CLI

Cara mengimpor kunci publik ke Amazon EC2

Gunakan perintah import-key-pair.

```
aws ec2 import-key-pair \
    --key-name my-key-pair \
    --public-key-material fileb://path/my-key-pair.pub
```

Cara melakukan verifikasi terhadap pasangan kunci yang berhasil diimpor

Gunakan perintah describe-key-pairs.

```
aws ec2 describe-key-pairs --key-names my-key-pair
```

PowerShell

Cara mengimpor kunci publik ke Amazon EC2

Gunakan Import-EC2KeyPair cmdlet.

```
$publickey=[Io.File]::ReadAllText("C:\Users\TestUser\.ssh\id_rsa.pub")
Import-EC2KeyPair `
    -KeyName my-key-pair `
    -PublicKey $publickey
```

Cara melakukan verifikasi terhadap pasangan kunci yang berhasil diimpor

Gunakan Get-EC2KeyPair cmdlet.

```
Get-EC2KeyPair -KeyName my-key-pair
```

## Jelaskan pasangan kunci Anda

Anda dapat menjelaskan pasangan kunci yang Anda simpan di Amazon EC2. Anda juga dapat mengambil materi kunci publik dan melakukan identifikasi terhadap kunci publik yang ditentukan saat peluncuran.

Tugas

- Jelaskan pasangan kunci Anda
- Mengambil materi kunci publik
- Mengidentifikasi kunci publik yang ditentukan saat peluncuran

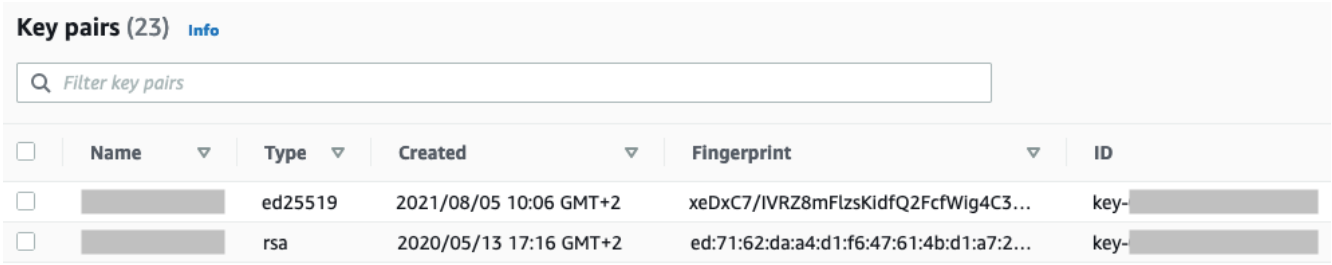
## Jelaskan pasangan kunci Anda

Anda dapat melihat informasi berikut ini tentang kunci publik Anda yang disimpan di Amazon EC2: nama kunci publik, ID, jenis kunci, sidik jari, materi kunci publik, tanggal dan waktu (di zona waktu UTC) tempat kunci dibuat oleh Amazon EC2 (jika kunci dibuat oleh alat pihak ketiga, maka tanggal dan waktu tersebut adalah tanggal dan waktu dari kunci yang diimpor ke Amazon EC2), dan setiap tanda yang dikaitkan dengan kunci publik tersebut.

Console

Untuk melihat informasi tentang pasangan kunci Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi yang ada di sebelah kiri, pilih Pasangan Kunci.
3. Anda dapat melihat informasi tentang setiap kunci publik dalam tabel Key Pairs (Pasangan kunci).




4. Untuk melihat tag kunci publik, pilih kotak centang di sebelah kunci, lalu pilih Tindakan, Kelola tag.

AWS CLI

Untuk melihat informasi tentang pasangan kunci

Gunakan perintah describe-key-pairs.

```
aws ec2 describe-key-pairs --key-names key-pair-name
```

PowerShell

Untuk melihat informasi tentang pasangan kunci

Gunakan Get-EC2KeyPair cmdlet.

```
Get-EC2KeyPair -KeyName key-pair-name
```

## Mengambil materi kunci publik

Anda bisa mendapatkan materi kunci publik untuk pasangan kunci Anda. Berikut ini adalah contoh kunci publik. Perhatikan bahwa ada jeda baris yang ditambahkan untuk keterbacaan.

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQClKsfkNkuSevGj3eYhCe53pcjqP3maAhDFcvBS7O6V
hz2ItxCih+PnDSUaw+WNQn/mZphTk/a/gU8jEzoOWbkM4yxyb/wB96xbiFveSFJuOp/d6RJhJOI0iBXr
lsLnBItntckiJ7FbtxJMXLvvwJryDUilBMTjYtwB+QhYXUMOzce5Pjz5/i8SeJtjnV3iAoG/cQk+0FzZ
qaeJAAHco+CY/5WrUBkrHmFJr6HcXkvJdWPkYQS3xqC0+FmUZofz221CBt5IMucxXPkX4rWi+z7wB3Rb
BQoQzd8v7yeb7OzlPnWOyN0qFU0XA246RA8QFYiCNYwI3f05p6KLxEXAMPLE
```

Private key

Untuk mengambil materi kunci publik menggunakan ssh-keygen (Linux)

Di komputer Linux atau macOS lokal Anda, gunakan ssh-keygen perintah. Tentukan jalur tempat Anda mengunduh kunci privat Anda (file `.pem`).

```
ssh-keygen -y -f /path_to_key_pair/my-key-pair.pem
```

Jika ssh-keygen perintah ini gagal, jalankan chmod perintah berikut untuk memastikan bahwa file kunci pribadi memiliki izin yang diperlukan.

```
chmod 400 key-pair-name.pem
```

Untuk mengambil materi kunci publik menggunakan PuttyGen (Windows)

Di komputer Windows lokal Anda, mulai PuttyGen. Pilih Muat. Pilih file kunci privat `.ppk` atau `.pem`. PuTTYgen menampilkan kunci publik di Kunci publik untuk ditempelkan ke file authorized_keys Open SSH. Anda juga dapat melihat kunci publik dengan memilih Simpan kunci publik, dengan menentukan nama untuk file, menyimpan file, lalu membuka file tersebut.

AWS CLI

Untuk mengambil materi kunci publik

Gunakan perintah describe-key-pair berikut dan tentukan opsi. `--include-public-key`

```
aws ec2 describe-key-pairs \
    --key-names key-pair-name \
    --include-public-key \
    --query "KeyPairs[].PublicKey"
```

PowerShell

Untuk mengambil materi kunci publik

Gunakan Get-EC2KeyPair cmdlet.

```
(Get-EC2KeyPair -KeyName key-pair-name -IncludePublicKey $true).PublicKey
```

IMDSv2

Linux

Jalankan perintah berikut dari instance Linux Anda.

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-
token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/public-keys/0/openssh-key
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda.

```
[string]$token = Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token-ttl-seconds"
 = "21600"} -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod -Headers @{"X-aws-ec2-metadata-token" = $token} -Method GET -Uri
 http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

IMDSv1

Linux

Jalankan perintah berikut dari instance Linux Anda.

```
curl http://169.254.169.254/latest/meta-data/public-keys/0/openssh-key
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda.

```
Invoke-RestMethod -uri  http://169.254.169.254/latest/meta-data/public-keys/0/
openssh-key
```

## Mengidentifikasi kunci publik yang ditentukan saat peluncuran

Jika Anda menentukan kunci publik saat meluncurkan instans, maka nama kunci publik tersebut akan direkam oleh instans. Nama kunci publik yang dilaporkan untuk instance tidak berubah, bahkan jika Anda mengubah kunci publik pada instans atau menambahkan kunci publik.

Console

Untuk mengidentifikasi kunci publik yang ditentukan saat peluncuran instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pada tab Detail, di bawah Detail Instance, temukan Pasangan kunci yang ditetapkan saat peluncuran.

AWS CLI

Untuk mengidentifikasi kunci publik yang ditentukan saat peluncuran instance

Gunakan perintah describe-instance berikut.

```
aws ec2 describe-instances \
    --instance-id i-1234567890abcdef0 \
    --query "Reservations[].Instances[].KeyName" \
    --output text
```

Berikut ini adalah output contoh.

```
key-pair-name
```

PowerShell

Untuk mengidentifikasi kunci publik yang ditentukan saat peluncuran instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -InstanceId i-1234567890abcdef0).Instances | Select KeyName
```

Berikut ini adalah output contoh.

```
KeyName
-------
key-pair-name
```

## Menghapus pasangan kunci Anda

Anda dapat menghapus pasangan kunci, yang menghapus kunci publik yang disimpan di Amazon EC2. Menghapus pasangan kunci tidak menghapus kunci pribadi yang cocok.

Saat Anda menghapus kunci publik menggunakan metode berikut, Anda hanya akan menghapus kunci publik yang Anda simpan di Amazon EC2 saat Anda membuat atau mengimpor pasangan kunci. Menghapus kunci publik tidak akan menghapus kunci publik tersebut dari instans mana pun yang padanya kunci publik tersebut telah Anda tambahkan, baik ketika Anda meluncurkan instans atau setelahnya. Hal ini juga tidak akan menghapus kunci privat di komputer lokal Anda. Anda dapat terus terhubung ke instans yang Anda luncurkan menggunakan kunci publik yang Anda hapus dari Amazon EC2 selama Anda masih memiliki file kunci privat (`.pem`).

> **Important**
>
> Jika Anda menggunakan grup Auto Scaling (misalnya, dalam lingkungan Elastic Beanstalk), pastikan bahwa kunci publik yang Anda hapus tidak ditentukan dalam templat peluncuran atau konfigurasi peluncuran yang dikaitkan. Jika Amazon EC2 Auto Scaling mendeteksi instans yang tidak sehat, ia akan meluncurkan instans pengganti. Namun demikian, peluncuran instans tersebut akan gagal jika kunci publik tidak dapat ditemukan. Untuk informasi selengkapnya, lihat Templat peluncuran di Panduan Pengguna Amazon EC2 Auto Scaling.

Console

Cara menghapus kunci publik Anda di Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Pasangan Kunci.
3. Pilih pasangan kunci yang akan dihapus lalu pilih Tindakan, Hapus.
4. Dalam bidang konfirmasi, masukkan `Delete` lalu pilih Delete (Hapus).

AWS CLI

Cara menghapus kunci publik Anda di Amazon EC2

Gunakan perintah delete-key-pair.

```
aws ec2 delete-key-pair --key-name my-key-pair
```

PowerShell

Cara menghapus kunci publik Anda di Amazon EC2

Gunakan Remove-EC2KeyPair cmdlet.

```
Remove-EC2KeyPair -KeyName my-key-pair
```

## Menambahkan atau mengganti kunci publik pada instans Linux Anda

| Deskripsi |
| --- |
| Jika Anda kehilangan kunci pribadi, Anda kehilangan akses ke instans apa pun yang menggunakan pasangan kunci. Untuk informasi selengkapnya tentang menghubungkan ke instance menggunakan pasangan kunci yang berbeda dari yang Anda tentukan saat peluncuran, lihat Saya kehilangan kunci pribadi saya. |

Saat meluncurkan instans, Anda dapat menentukan pasangan kunci. Jika Anda menentukan pasangan kunci, saat instans Anda melakukan booting untuk pertama kalinya, materi kunci publik

akan ditempatkan pada instans Linux Anda di entri dalam `~/.ssh/authorized_keys`. Ketika Anda pertama kali terhubung ke instance Linux menggunakan SSH, Anda menentukan pengguna default dan kunci pribadi yang sesuai dengan kunci publik yang disimpan di instance Linux Anda.

Setelah terhubung ke instans, Anda dapat mengubah pasangan kunci yang digunakan untuk mengakses akun sistem default instans Anda dengan menambahkan kunci publik baru pada instans, atau dengan mengganti kunci publik (menghapus kunci publik yang ada dan menambahkan yang baru) pada instans. Anda juga dapat menghapus semua kunci publik dari instans.

Anda dapat menambahkan atau mengganti pasangan kunci karena alasan berikut:

- Jika seorang pengguna dalam organisasi Anda memerlukan akses ke akun pengguna sistem yang menggunakan pasangan kunci terpisah, maka Anda dapat menambahkan kunci publik tersebut ke instans Anda.
- Jika seseorang memiliki salinan kunci privat (file `.pem`) dan Anda ingin mencegahnya agar tidak terhubung ke instans Anda (misalnya, ketika orang tersebut sudah meninggalkan organisasi Anda), maka Anda dapat mengganti kunci publik dengan yang baru.
- Jika Anda membuat AMI Linux dari instans, materi kunci publik akan disalin dari instans ke AMI. Jika Anda meluncurkan instans dari AMI, maka instans baru tersebut akan menyertakan kunci publik dari instans asli. Untuk mencegah seseorang yang memiliki kunci privat terhubung ke instans yang baru, Anda dapat menghapus kunci publik dari instans asli sebelum membuat AMI.

Gunakan prosedur berikut untuk memodifikasi pasangan kunci untuk pengguna default, sepertiec2-user. Untuk informasi tentang menambahkan pengguna ke instans Anda, lihat dokumentasi untuk sistem operasi pada instans Anda.

Cara menambah atau mengganti pasangan kunci

1. Buatlah pasangan kunci baru menggunakan konsol Amazon EC2 atau alat pihak ketiga.
2. Ambil kunci publik dari pasangan kunci baru Anda. Untuk informasi selengkapnya, lihat Mengambil materi kunci publik.
3. Terhubung ke instans Anda.
4. Pada contoh, menggunakan editor teks pilihan Anda, buka `.ssh/authorized_keys` file. Tempelkan informasi kunci publik dari pasangan kunci baru Anda di bawah informasi kunci publik yang ada dan kemudian simpan file tersebut.
5. Putuskan sambungan dari instance Anda. Uji apakah Anda dapat terhubung ke instans menggunakan file kunci pribadi dari pasangan kunci baru.

6. Jika Anda menggunakan Penskalaan Otomatis, armada EC2, atau template peluncuran untuk meluncurkan instans, periksa apakah pasangan kunci yang Anda ganti ditentukan dalam template peluncuran atau konfigurasi peluncuran. Jika tidak, peluncuran instance akan gagal.
7. (Opsional) Jika Anda mengganti pasangan kunci yang sudah ada, hubungkan ke instans Anda dan hapus informasi kunci publik untuk pasangan kunci asli dari file `.ssh/authorized_keys`.

Untuk menghapus kunci publik dari instans

1. Terhubung ke instans Anda.
2. Dengan menggunakan editor teks yang Anda kehendaki, buka file `.ssh/authorized_keys` pada instans. Hapus informasi kunci publik, dan kemudian simpan file.

> **Warning**
>
> Jika Anda menghapus semua kunci publik dari instans dan memutuskan sambungan dari instans, Anda tidak dapat menyambung ke instans lagi kecuali Anda telah mengonfigurasi cara alternatif untuk masuk.

## Lakukan verifikasi terhadap sidik jari pasangan kunci Anda

Untuk memverifikasi sidik jari pasangan kunci Anda, bandingkan sidik jari yang ditampilkan pada halaman Pasangan kunci di konsol Amazon EC2, atau dikembalikan oleh perintah describe-key-pairs, dengan sidik jari yang Anda hasilkan menggunakan kunci privat di komputer lokal Anda. Sidik jari ini harus cocok.

Saat Amazon EC2 menghitung sidik jari, Amazon EC2 mungkin menambahkan padding ke sidik jari dengan karakter =. Alat lain, seperti ssh-keygen, mungkin menghilangkan padding ini.

Jika Anda mencoba memverifikasi sidik jari instans Linux EC2, bukan sidik jari pasangan kunci Anda, lihat Mendap atkan sidik jari instans.

### Cara sidik jari dikalkulasi

Amazon EC2 menggunakan berbagai fungsi hash untuk menghitung sidik jari untuk pasangan kunci RSA dan ED25519. Selain itu, untuk pasangan kunci RSA, Amazon EC2 menghitung sidik jari secara berbeda menggunakan fungsi hash yang berbeda tergantung pada apakah pasangan kunci tersebut dibuat oleh Amazon EC2 atau diimpor ke Amazon EC2.

Tabel berikut ini mencantumkan fungsi hash yang digunakan untuk menghitung sidik jari untuk pasangan kunci RSA dan ED25519 yang dibuat oleh Amazon EC2 dan diimpor ke Amazon EC2.

(Instans Linux) Fungsi hash digunakan untuk menghitung sidik jari

| Sumber pasangan kunci | Pasangan kunci RSA (Windows dan Linux) | Pasangan kunci ED25519 (Linux) |
|---|---|---|
| Dibuat oleh Amazon EC2 | SHA-1 | SHA-256 |
| Diimpor ke Amazon EC2 | MD5[1] | SHA-256 |

[1] Jika Anda mengimpor kunci RSA publik ke Amazon EC2, sidik jari akan dihitung menggunakan fungsi hash MD5. Hal ini berlaku terlepas dari cara yang Anda gunakan untuk membuat pasangan kunci, misalnya, menggunakan alat pihak ketiga atau dengan membuat kunci publik baru dari kunci privat yang sudah ada yang dibuat menggunakan Amazon EC2.

## Saat menggunakan pasangan kunci yang sama di Wilayah yang berbeda

Jika Anda berencana menggunakan pasangan kunci yang sama untuk terhubung ke instans yang berbeda Wilayah AWS, Anda harus mengimpor kunci publik ke semua Wilayah tempat Anda akan menggunakannya. Jika Anda menggunakan Amazon EC2 untuk membuat pasangan kunci, Anda dapat Mengambil materi kunci publik mengimpor kunci publik ke Wilayah lain.

> **Note**
>
> - Jika Anda membuat pasangan kunci RSA menggunakan Amazon EC2, dan kemudian Anda membuat kunci publik dari kunci privat Amazon EC2, kunci publik yang diimpor akan memiliki sidik jari yang berbeda dari kunci publik yang asli. Ini karena sidik jari kunci RSA asli yang dibuat menggunakan Amazon EC2 dihitung menggunakan fungsi SHA-1 hash, sedangkan sidik jari kunci RSA yang diimpor dihitung menggunakan fungsi hash MD5.
> - Untuk pasangan kunci ED25519, sidik jari akan sama terlepas dari apakah mereka dibuat oleh Amazon EC2 atau diimpor ke Amazon EC2, karena fungsi SHA-256 hash yang sama digunakan untuk menghitung sidik jari.

## Membuat sidik jari dari kunci privat

Gunakan salah satu perintah berikut untuk membuat sidik jari dari kunci privat di mesin lokal Anda.

Jika Anda menggunakan mesin lokal Windows, maka Anda dapat menjalankan perintah berikut menggunakan Windows Subsystem for Linux (WSL). Instal WSL dan distribusi Linux menggunakan instruksi di Cara menginstal Linux di Windows dengan W SL. Contoh dalam instruksi tersebut menginstal distribusi Ubuntu Linux, tetapi Anda dapat menginstal distribusi apa pun. Anda akan diminta untuk memulai ulang komputer Anda agar perubahan dapat diterapkan.

- Jika Anda membuat pasangan kunci menggunakan Amazon EC2

  Gunakan alat OpenSSL untuk membuat sidik jari seperti yang ditunjukkan dalam contoh-contoh berikut.

  Untuk pasangan kunci RSA:

  ```
  openssl pkcs8 -in path_to_private_key -inform PEM -outform DER -topk8 -nocrypt |
   openssl sha1 -c
  ```

  (Instans Linux) Untuk pasangan kunci ED25519:

  ```
  ssh-keygen -l -f path_to_private_key
  ```

- (Hanya pasangan kunci RSA) Jika Anda mengimpor kunci publik ke Amazon EC2

  Anda dapat mengikuti prosedur ini terlepas dari cara Anda membuat pasangan kunci, misalnya menggunakan alat pihak ketiga atau dengan membuat kunci publik baru dari kunci privat yang sudah ada yang dibuat menggunakan Amazon EC2

  Gunakan alat OpenSSL untuk membuat sidik jari seperti yang ditunjukkan dalam contoh berikut.

  ```
  openssl rsa -in path_to_private_key -pubout -outform DER | openssl md5 -c
  ```

- Jika Anda membuat pasangan kunci OpenSSH menggunakan OpenSSH 7.8 atau yang lebih baru dan mengimpor kunci publik ke Amazon EC2

  Gunakan ssh-keygen untuk membuat sidik jari seperti yang ditunjukkan dalam contoh-contoh berikut.

  Untuk pasangan kunci RSA:

```
ssh-keygen -ef path_to_private_key -m PEM | openssl rsa -RSAPublicKey_in -outform DER
 | openssl md5 -c
```

(Instans Linux) Untuk pasangan kunci ED25519:

```
ssh-keygen -l -f path_to_private_key
```

# Grup keamanan Amazon EC2 untuk instans EC2 Anda

grup keamanan bertindak sebagai firewall virtual untuk instans EC2 Anda untuk mengontrol lalu lintas masuk dan ke luar. Aturan-aturan ke dalam mengontrol lalu lintas yang masuk ke instans Anda, dan aturan-aturan ke luar mengontrol lalu lintas yang ke luar dari instans Anda. Saat Anda meluncurkan instans, artinya Anda menentukan satu atau beberapa grup keamanan pada instans tersebut. Jika Anda tidak menentukan grup keamanan, maka Amazon EC2 akan menggunakan grup keamanan default untuk VPC. Setelah Anda meluncurkan instans, Anda dapat mengubah grup keamanannya.

Keamanan adalah tanggung jawab bersama AWS antara Anda. Untuk informasi lebih lanjut, lihatKeamanan dalam Amazon EC2. AWS menyediakan grup keamanan sebagai salah satu alat untuk mengamankan instance Anda, dan Anda perlu mengonfigurasinya untuk memenuhi kebutuhan keamanan Anda. Jika Anda memiliki persyaratan yang tidak sepenuhnya dipenuhi oleh grup keamanan, maka Anda dapat mempertahankan firewall Anda sendiri pada instans Anda selain menggunakan grup keamanan.

Harga

Tidak ada biaya tambahan untuk menggunakan grup keamanan.

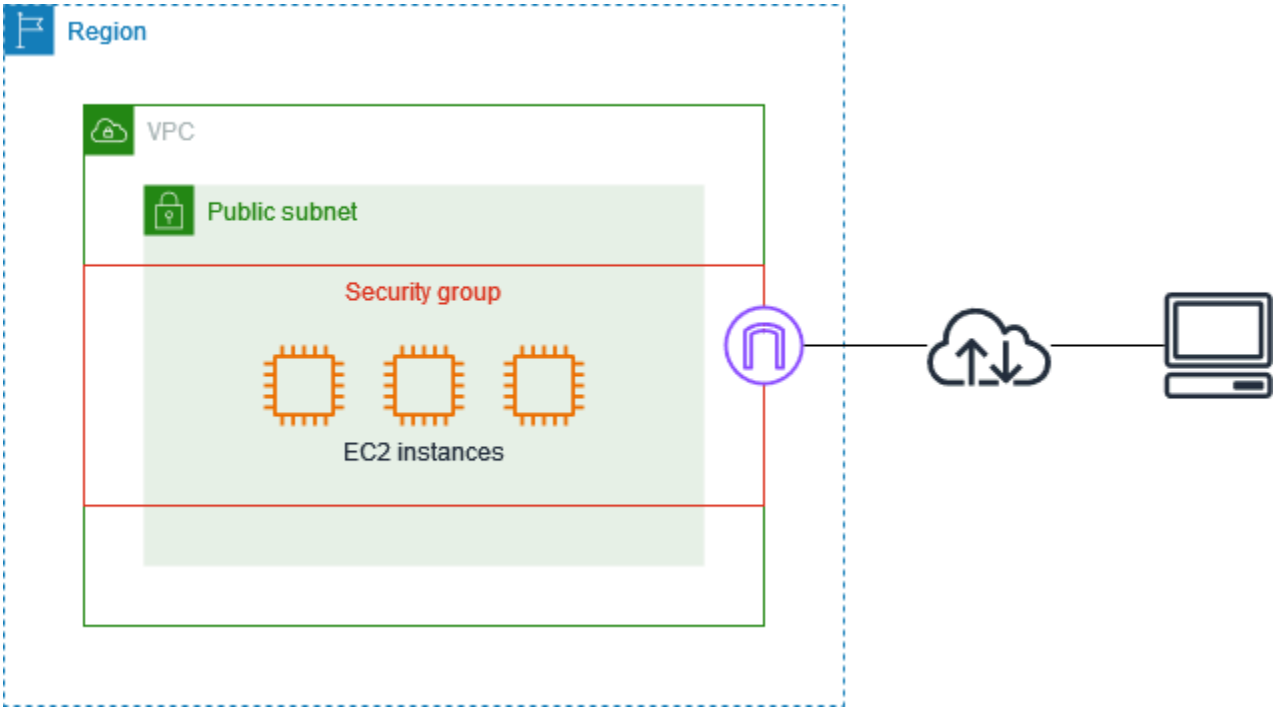
Daftar Isi


- Gambaran umum
- Membuat grup keamanan untuk instans Amazon EC2 Anda
- Mengubah grup keamanan untuk instans Amazon EC2 Anda
- Menghapus grup keamanan Amazon EC2
- Pelacakan koneksi grup keamanan Amazon EC2
- Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda

# Gambaran umum

Anda dapat mengaitkan setiap instance dengan beberapa grup keamanan, dan Anda dapat mengaitkan setiap grup keamanan dengan beberapa instans. Anda menambahkan aturan ke setiap grup keamanan yang mengizinkan lalu lintas ke atau dari instans terkait. Anda dapat melakukan modifikasi terhadap aturan-aturan untuk grup keamanan kapan saja. Aturan-aturan baru dan aturan-aturan yang dimodifikasi akan secara otomatis diterapkan ke semua instans yang dikaitkan dengan grup keamanan. Saat Amazon EC2 memutuskan apakah akan mengizinkan lalu lintas mencapai instans, Amazon akan mengevaluasi semua aturan dari semua grup keamanan yang terkait dengan instans tersebut. Untuk informasi selengkapnya, lihat Aturan grup keamanan di Panduan Pengguna Amazon VPC.

Diagram berikut menunjukkan VPC dengan subnet, gateway internet, dan grup keamanan. Subnet berisi instance EC2. Grup keamanan dikaitkan dengan instance. Satu-satunya lalu lintas yang mencapai instance adalah lalu lintas yang diizinkan oleh aturan grup keamanan. Misalnya, jika grup keamanan berisi aturan yang mengizinkan lalu lintas SSH dari jaringan Anda, maka Anda dapat terhubung ke instance dari komputer menggunakan SSH. Jika grup keamanan berisi aturan yang mengizinkan semua lalu lintas dari sumber daya yang terkait dengannya, maka setiap instance dapat menerima lalu lintas apa pun yang dikirim dari instance lain.




Grup keamanan bersifat stateful—jika Anda mengirimkan permintaan dari instans Anda, maka lalu lintas tanggapan untuk permintaan tersebut diperbolehkan untuk mengalir tanpa memedulikan

aturan-aturan ke dalam grup keamanan. Selain itu, respons terhadap lalu lintas masuk yang diizinkan diizinkan mengalir keluar, terlepas dari aturan keluar. Untuk informasi selengkapnya, lihat Pelacakan koneksi.

## Membuat grup keamanan untuk instans Amazon EC2 Anda

Grup keamanan bertindak sebagai firewall untuk instans-instans yang dikaitkan, mengontrol lalu lintas ke dalam dan ke luar pada tingkat instans. Anda dapat menambahkan aturan ke grup keamanan yang memungkinkan Anda untuk terhubung ke instans menggunakan SSH (instans Linux) atau RDP (instans Windows). Anda juga dapat menambahkan aturan yang memungkinkan lalu lintas klien, misalnya lalu lintas HTTP dan HTTPS yang ditujukan ke server web.

Anda dapat mengaitkan grup keamanan dengan instance saat meluncurkan instans. Saat Anda menambahkan atau menghapus aturan dari grup keamanan terkait, perubahan tersebut secara otomatis diterapkan ke semua instans yang telah Anda kaitkan grup keamanan.

Setelah meluncurkan instance, Anda dapat mengaitkan grup keamanan tambahan. Untuk informasi selengkapnya, lihat Mengubah grup keamanan untuk instans Amazon EC2 Anda.

Anda dapat menambahkan aturan grup keamanan masuk dan keluar saat membuat grup keamanan atau Anda dapat menambahkannya nanti. Untuk informasi selengkapnya, lihat Mengonfigurasi aturan grup keamanan. Untuk contoh aturan yang dapat Anda tambahkan ke grup keamanan, lihatAturan-aturan grup keamanan untuk kasus penggunaan yang berbeda.

Pertimbangan-pertimbangan

- Grup keamanan baru dimulai dengan hanya aturan keluar yang memungkinkan semua lalu lintas meninggalkan sumber daya. Anda harus menambahkan aturan-aturan lain untuk mengizinkan lalu lintas ke dalam atau membatasi lalu lintas ke luar.
- Saat mengonfigurasi sumber untuk aturan yang memungkinkan akses SSH atau RDP ke instance Anda, jangan izinkan akses dari mana saja, karena itu akan memungkinkan akses ini ke instance Anda dari semua alamat IP di internet. Hal ini dapat diterima untuk waktu yang singkat di lingkungan pengujian, tetapi tidak aman untuk lingkungan produksi.
- Jika ada lebih dari satu aturan untuk port tertentu, maka Amazon EC2 akan menerapkan aturan yang paling permisif. Misalnya, jika Anda memiliki aturan yang memungkinkan akses ke port TCP 22 (SSH) dari alamat IP 203.0.113.1, dan aturan lain yang memungkinkan akses ke port TCP 22 dari mana saja, maka setiap orang memiliki akses ke port TCP 22.
- Anda dapat mengaitkan beberapa grup keamanan dengan sebuah instance. Oleh karena itu, instans dapat memiliki ratusan aturan yang berlaku. Hal ini dapat menyebabkan masalah saat

Anda mengakses instans tersebut. Kami menyarankan agar Anda sedapat mungkin membuat aturan-aturan yang padat.

- Saat Anda menentukan grup keamanan sebagai sumber atau tujuan dari aturan, aturan tersebut akan memengaruhi semua instans yang dikaitkan dengan grup keamanan tersebut. Lalu lintas masuk diizinkan berdasarkan alamat IP privat dari instans yang dikaitkan dengan grup keamanan sumber (dan bukan alamat IP publik atau alamat IP Elastis). Untuk informasi selengkapnya tentang alamat IP, lihat Pengalamatan IP instans Amazon EC2.
- Amazon EC2 memblokir lalu lintas di port 25 secara default. Untuk informasi selengkapnya, lihat Pembatasan pada email yang dikirim menggunakan port 25.

Console

Cara membuat grup keamanan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Security Groups (Grup Keamanan).
3. Pilih Buat grup keamanan.
4. Masukkan nama deskriptif dan deskripsi singkat untuk grup keamanan. Anda tidak dapat mengubah nama dan deskripsi grup keamanan setelah dibuat.
5. Untuk VPC, pilih VPC tempat Anda akan menjalankan instans Amazon EC2 Anda.
6. (Opsional) Untuk menambahkan aturan masuk, pilih aturan masuk. Untuk setiap aturan, pilih Tambahkan aturan dan tentukan protokol, port, dan sumber. Misalnya, untuk mengizinkan lalu lintas SSH, pilih SSH untuk Jenis dan tentukan alamat IPv4 publik komputer atau jaringan Anda untuk Sumber.
7. (Opsional) Untuk menambahkan aturan keluar, pilih Aturan keluar. Untuk setiap aturan, pilih Tambahkan aturan dan tentukan protokol, port, dan tujuan. Jika tidak, Anda dapat mempertahankan aturan default, yang memungkinkan semua lalu lintas keluar.
8. (Opsional) Untuk menambahkan tag, pilih Tambahkan tag baru dan masukkan kunci tag dan nilai.
9. Pilih Create security group (Buat grup keamanan).

AWS CLI

Cara membuat grup keamanan

Gunakan perintah create-security-group berikut ini.

```
aws ec2 create-security-group \
    --group-name my-security-group \
    --description "my security group" \
    --vpc-id vpc-1234567890abcdef0
```

Untuk contoh yang menambahkan aturan, lihatthe section called “Mengonfigurasi aturan grup keamanan”.

PowerShell

Cara membuat grup keamanan

Gunakan New-EC2SecurityGroup cmdlet.

```
New-EC2SecurityGroup `
    -GroupName my-security-group `
    -Description "my security group" `
    -VpcId vpc-1234567890abcdef0
```

Untuk contoh yang menambahkan aturan, lihatthe section called “Mengonfigurasi aturan grup keamanan”.

## Mengubah grup keamanan untuk instans Amazon EC2 Anda

Anda dapat menentukan grup keamanan untuk instans Amazon EC2 saat Anda meluncurkannya. Setelah meluncurkan instance, Anda dapat menambahkan atau menghapus grup keamanan. Anda juga dapat menambahkan, menghapus, atau mengedit aturan grup keamanan untuk grup keamanan terkait kapan saja.

Grup keamanan dikaitkan dengan antarmuka jaringan. Menambahkan atau menghapus grup keamanan mengubah grup keamanan yang terkait dengan antarmuka jaringan utama. Anda juga dapat mengubah grup keamanan yang terkait dengan antarmuka jaringan sekunder. Untuk informasi selengkapnya, lihat Memodifikasi atribut antarmuka jaringan.

Tugas

- Menambahkan atau menghapus grup keamanan
- Mengonfigurasi aturan grup keamanan

## Menambahkan atau menghapus grup keamanan

Setelah meluncurkan instance, Anda dapat menambahkan atau menghapus grup keamanan dari daftar grup keamanan terkait. Saat Anda mengaitkan beberapa grup keamanan dengan instans, aturan-aturan dari masing-masing grup keamanan akan digabungkan secara efektif untuk membuat satu set aturan. Amazon EC2 menggunakan seperangkat aturan ini untuk menentukan apakah akan mengizinkan lalu lintas.

### Persyaratan

- Instans harus berada dalam status `running` atau `stopped`.

### Console

Untuk mengubah grup keamanan instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans Anda, dan kemudian pilih Actions (Tindakan), Security (Keamanan), Change security groups (Ubah grup keamanan).
4. Untuk Grup keamanan terkait, pilih grup keamanan dari daftar dan pilih Add security group (Tambahkan grup keamanan).

   Untuk menghapus grup keamanan yang sudah dikaitkan, pilih Remove (Hapus) untuk grup keamanan itu.
5. Pilih Save (Simpan).

### AWS CLI

Untuk mengubah grup keamanan instans

Gunakan perintah modify- instance-atribut berikut.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --groups sg-1234567890abcdef0
```

PowerShell

Untuk mengubah grup keamanan instans

Gunakan Edit-EC2InstanceAttribute cmdlet.

```
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -Group sg-1234567890abcdef0
```

## Mengonfigurasi aturan grup keamanan

Setelah membuat grup keamanan, Anda dapat menambahkan, memperbarui, dan menghapus aturan grup keamanannya. Saat Anda menambahkan, memperbarui, atau menghapus aturan, perubahan secara otomatis diterapkan ke sumber daya yang terkait dengan grup keamanan.

Untuk contoh aturan yang dapat Anda tambahkan ke grup keamanan, lihatAturan-aturan grup keamanan untuk kasus penggunaan yang berbeda.

Izin yang diperlukan

Sebelum Anda mulai, pastikan Anda memiliki izin yang diperlukan. Untuk informasi selengkapnya, lihat Contoh: Cara menggunakan grup keamanan.

Protokol dan Port

- Dengan konsol, ketika Anda memilih jenis yang telah ditentukan, jang kauan Protokol dan Port ditentukan untuk Anda. Untuk memasukkan rentang port, Anda harus memilih salah satu jenis kustom berikut: C ustom TCP atau C ustom UD P.
- Dengan AWS CLI, Anda dapat menambahkan aturan tunggal dengan satu port menggunakan `--port` opsi `--protocol` dan. Untuk menambahkan beberapa aturan, atau aturan dengan rentang port, gunakan `--ip-permissions` opsi sebagai gantinya.

Sumber dan tujuan

- Dengan konsol, Anda dapat menentukan berikut ini sebagai sumber untuk aturan masuk atau tujuan untuk aturan keluar:
  - Kustom — Blok IPv4 CIDR, blok IPv6 CIDR, grup keamanan, atau daftar awalan.

- Anywhere-IPv4— 0.0.0. 0/0 Blok IPv4 CIDR.
- Anywhere-IPv6- Blok CIDR IPv6: :/0.
- IP Saya — Alamat IPv4 publik komputer lokal Anda.

- Dengan AWS CLI, Anda dapat menentukan blok IPv4 CIDR menggunakan `--cidr` opsi atau grup keamanan menggunakan opsi. `--source-group` Untuk menentukan daftar awalan atau blok CIDR IPv6, gunakan opsi. `--ip-permissions`

> **Warning**
>
> Jika Anda menambahkan aturan masuk untuk port 22 (SSH) atau 3389 (RDP), kami sangat menyarankan Anda untuk mengotorisasi hanya alamat IP tertentu atau rentang alamat yang memerlukan akses ke instans Anda. Jika Anda memilih Anywhere-IPv4, Anda mengizinkan lalu lintas dari semua alamat IPv4 untuk mengakses instance Anda menggunakan protokol yang ditentukan. Jika Anda memilih Anywhere-IPv6, Anda mengizinkan lalu lintas dari semua alamat IPv6 untuk mengakses instance Anda menggunakan protokol yang ditentukan.

Console

Mengonfigurasi aturan grup keamanan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Security Groups (Grup Keamanan).
3. Pilih grup keamanan.
4. Untuk mengedit aturan masuk, pilih Edit aturan masuk dari Tindakan atau tab Aturan masuk.

   a. Untuk menambahkan aturan, pilih Tambahkan aturan dan masukkan jenis, protokol, port, dan sumber untuk aturan tersebut.

      Jika jenisnya TCP atau UDP, Anda harus memasukkan rentang port untuk mengizinkan. Untuk ICMP kustom, Anda harus memilih jenis ICMP dari Protokol, dan, jika memungkinkan, nama kode dari Rentang port. Untuk jenis lainnya, protokol dan rentang port akan dikonfigurasikan untuk Anda.

   b. Untuk memperbarui aturan, ubah protokol, deskripsi, dan sumbernya sesuai kebutuhan. Namun, Anda tidak dapat mengubah jenis sumber. Misalnya, jika sumbernya adalah

blok CIDR IPv4, Anda tidak dapat menentukan blok CIDR IPv6, daftar awalan, atau grup keamanan.

c. Untuk menghapus aturan, pilih tombol Delete.

5. Untuk mengedit aturan keluar, pilih Edit aturan keluar dari Tindakan atau tab Aturan keluar.

   a. Untuk menambahkan aturan, pilih Tambah aturan dan masukkan jenis, protokol, port, dan tujuan untuk aturan tersebut. Anda juga dapat memasukkan deskripsi opsional.

      Jika jenisnya TCP atau UDP, Anda harus memasukkan rentang port untuk mengizinkan. Untuk ICMP kustom, Anda harus memilih jenis ICMP dari Protokol, dan, jika memungkinkan, nama kode dari Rentang port. Untuk jenis lainnya, protokol dan rentang port akan dikonfigurasikan untuk Anda.

   b. Untuk memperbarui aturan, ubah protokol, deskripsi, dan sumbernya sesuai kebutuhan. Namun, Anda tidak dapat mengubah jenis sumber. Misalnya, jika sumbernya adalah blok CIDR IPv4, Anda tidak dapat menentukan blok CIDR IPv6, daftar awalan, atau grup keamanan.

   c. Untuk menghapus aturan, pilih tombol Delete.

6. Pilih Simpan aturan.

AWS CLI

Untuk menambahkan aturan grup keamanan

Gunakan perintah authorize-security-group-ingress untuk menambahkan aturan masuk. Contoh berikut memungkinkan lalu lintas SSH masuk dari blok CIDR dalam daftar awalan yang ditentukan.

```
aws ec2 authorize-security-group-ingress \
    --group-id sg-1234567890abcdef0 \
    --ip-permissions
 'IpProtocol=tcp,FromPort=22,ToPort=22,PrefixListIds=[{PrefixListId=pl-
f8a6439156EXAMPLE}]'
```

Gunakan perintah authorize-security-group-egress untuk menambahkan aturan keluar. Contoh berikut memungkinkan lalu lintas TCP keluar pada port 80 ke instance dengan grup keamanan yang ditentukan.

```
aws ec2 authorize-security-group-egress \
    --group-id sg-1234567890abcdef0 \
    --ip-permissions
 'IpProtocol=tcp,FromPort=80,ToPort=80,UserIdGroupPairs=[{GroupId=sg-0aad1c26bb6EXAMPLE}]'
```

Untuk menghapus aturan grup keamanan

Gunakan perintah revoke-security-group-Ingress berikut untuk menghapus aturan masuk.

```
aws ec2 revoke-security-group-egress \
    --group id sg-1234567890abcdef0 \
    --security-group-rule-ids sgr-09ed298024EXAMPLE
```

Gunakan perintah revoke-security-group-eg ress berikut untuk menghapus aturan keluar.

```
aws ec2 revoke-security-group-ingress \
    --group id sg-1234567890abcdef0 \
    --security-group-rule-ids sgr-0352250c1aEXAMPLE
```

Untuk mengubah aturan grup keamanan

Gunakan perintah modify- security-group-rules. Contoh berikut mengubah blok CIDR IPv4 dari aturan grup keamanan yang ditentukan.

```
aws ec2 modify-security-group-rules \
    --group id sg-1234567890abcdef0 \
    --security-group-rules
 'SecurityGroupRuleId=sgr-09ed298024EXAMPLE,SecurityGroupRule={IpProtocol=tcp,FromPort=80,To
```

PowerShell

Untuk menambahkan aturan grup keamanan

Gunakan Grant-EC2SecurityGroupIngress cmdlet untuk menambahkan aturan masuk. Contoh berikut memungkinkan lalu lintas SSH masuk dari blok CIDR dalam daftar awalan yang ditentukan.

```
$plid = New-Object -TypeName Amazon.EC2.Model.PrefixListId
$plid.Id = "pl-f8a6439156EXAMPLE"
Grant-EC2SecurityGroupIngress `
```

```
    -GroupId sg-1234567890abcdef0 `
    -IpPermission @{IpProtocol="tcp"; FromPort=22; ToPort=22; PrefixListIds=$plid}
```

Gunakan Grant-EC2SecurityGroupEgress cmdlet untuk menambahkan aturan keluar. Contoh berikut memungkinkan lalu lintas TCP keluar pada port 80 ke instance dengan grup keamanan yang ditentukan.

```
$uigp = New-Object -TypeName Amazon.EC2.Model.UserIdGroupPair
$uigp.GroupId = "sg-0aad1c26bb6EXAMPLE"
Grant-EC2SecurityGroupEgress `
    -GroupId sg-1234567890abcdef0 `
    -IpPermission @{IpProtocol="tcp"; FromPort=80; ToPort=80; UserIdGroupPairs=
$uigp}
```

Untuk menghapus aturan grup keamanan

Gunakan Revoke-EC2SecurityGroupIngress cmdlet untuk menghapus aturan masuk.

```
Revoke-EC2SecurityGroupIngress `
    -GroupId sg-1234567890abcdef0 `
    -SecurityGroupRuleId sgr-09ed298024EXAMPLE
```

Gunakan Revoke-EC2SecurityGroupEgress cmdlet untuk menghapus aturan keluar.

```
Revoke-EC2SecurityGroupEgress `
    -GroupId sg-1234567890abcdef0 `
    -SecurityGroupRuleId sgr-0352250c1aEXAMPLE
```

Untuk mengubah aturan grup keamanan

Gunakan Edit-EC2SecurityGroupRule cmdlet. Contoh berikut mengubah blok CIDR IPv4 dari aturan grup keamanan yang ditentukan.

```
$sgrr = New-Object -TypeName Amazon.EC2.Model.SecurityGroupRuleRequest
$sgrr.IpProtocol = "tcp"
$sgrr.FromPort = 80
$sgrr.ToPort = 80
$sgrr.CidrIpv4 = "0.0.0.0/0"
$sgr = New-Object -TypeName Amazon.EC2.Model.SecurityGroupRuleUpdate
$sgr.SecurityGroupRuleId = "sgr-09ed298024EXAMPLE"
```

```
$sgr.SecurityGroupRule = $sgrr
Edit-EC2SecurityGroupRule  `
    -GroupId sg-1234567890abcdef0 `
    -SecurityGroupRule $sgr
```

## Menghapus grup keamanan Amazon EC2

Setelah selesai dengan grup keamanan yang Anda buat untuk digunakan dengan instans Amazon EC2, Anda dapat menghapusnya.

Persyaratan

- Grup keamanan tidak dapat dikaitkan dengan instance atau antarmuka jaringan.
- Grup keamanan tidak dapat direferensikan oleh aturan di grup keamanan lain.

Console

Cara menghapus grup keamanan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. (Opsional) Untuk memverifikasi bahwa grup keamanan Anda tidak terkait dengan instance, lakukan hal berikut:
   a. Pada panel navigasi, pilih Grup Keamanan.
   b. Salin ID grup keamanan untuk dihapus.
   c. Di panel navigasi, pilih Instans.
   d. Di bilah pencarian, tambahkan ID grup keamanan sama dengan filter dan tempel ID grup keamanan. Jika tidak ada hasil, maka grup keamanan tidak terkait dengan instance. Jika tidak, Anda harus memutuskan hubungan grup keamanan sebelum Anda dapat menghapusnya.
3. Pada panel navigasi, pilih Grup Keamanan.
4. Pilih grup keamanan lalu pilih Tindakan, Hapus Grup Keamanan.
5. Jika Anda memilih lebih dari satu grup keamanan, Anda akan diminta untuk konfirmasi. Jika beberapa grup keamanan tidak dapat dihapus, kami menampilkan status setiap grup keamanan, yang menunjukkan apakah grup tersebut akan dihapus. Untuk mengonfirmasi penghapusan, masukkan Delete.

6. Pilih Hapus.

AWS CLI

Cara menghapus grup keamanan

Gunakan perintah delete-security-group berikut ini.

```
aws ec2 delete-security-group --group-id sg-1234567890abcdef0
```

PowerShell

Cara menghapus grup keamanan

Gunakan Remove-EC2SecurityGroup cmdlet.

```
Remove-EC2SecurityGroup -GroupId sg-1234567890abcdef0
```

## Pelacakan koneksi grup keamanan Amazon EC2

Grup keamanan Anda menggunakan pelacakan koneksi untuk melacak informasi tentang lalu lintas ke dan dari instans. Aturan-aturan diterapkan berdasarkan status koneksi lalu lintas untuk menentukan apakah lalu lintas diizinkan atau ditolak. Dengan pendekatan ini, grup keamanan berada dalam status stateful. Artinya tanggapan-tanggapan terhadap lalu lintas ke dalam diizinkan mengalir ke luar dari instans tanpa memedulikan aturan grup keamanan ke luar, dan sebaliknya.

Sebagai contoh, anggaplah bahwa Anda memulai perintah seperti netcat atau yang mirip dengan instans Anda dari komputer rumah Anda, dan aturan grup keamanan ke dalam Anda mengizinkan lalu lintas ICMP. Informasi tentang koneksi (termasuk informasi port) akan dilacak. Lalu lintas tanggapan dari instans untuk perintah tidak dilacak sebagai permintaan baru, tetapi sebagai koneksi yang telah terbentuk dan diizinkan untuk mengalir ke luar dari instans, meskipun aturan grup keamanan ke luar Anda membatasi lalu lintas ICMP ke luar.

Untuk protokol selain TCP, UDP, atau ICMP, hanya alamat IP dan nomor protokol saja yang dilacak. Jika instans Anda mengirimkan lalu lintas ke host lain, dan host tersebut mengirimkan jenis lalu lintas yang sama ke instans Anda dalam 600 detik, maka grup keamanan untuk instans Anda akan menerimanya terlepas dari aturan-aturan ke dalam grup keamanan tersebut. Grup keamanan tersebut menerimanya karena dianggap sebagai lalu lintas tanggapan untuk lalu lintas asli.

Ketika Anda mengubah aturan grup keamanan, koneksi-koneksi yang dilacak tidak akan langsung terputus. Grup keamanan akan tetap mengizinkan paket sampai koneksi yang ada waktunya habis. Untuk memastikan lalu lintas langsung terputus, atau bahwa semua lalu lintas tunduk pada aturan-aturan firewall tanpa memedulikan status pelacakan, Anda dapat menggunakan ACL jaringan untuk subnet Anda. ACL jaringan bersifat stateless dan karenanya tidak akan mengizinkan lalu lintas tanggapan secara otomatis. Menambahkan ACL jaringan yang memblokir lalu lintas di salah satu arah akan memutuskan koneksi yang ada. Untuk informasi selengkapnya, lihat ACL Jaringan di Panduan Pengguna Amazon VPC.

> **Note**
>
> Grup keamanan tidak berpengaruh pada lalu lintas DNS ke atau dari Route 53 Resolver, kadang-kadang disebut sebagai 'alamat IP VPC+2' (lihat Apa itu Amazon Route 53 Resolver? di Panduan Pengembang Amazon Route 53), atau 'AmazonProvidedDNS' (lihat Bek erja dengan set opsi DHCP di Panduan Pengguna Amazon Virtual Private Cloud). Jika Anda ingin memfilter permintaan DNS melalui Route 53 Resolver, Anda dapat mengaktifkan Firewall DNS Route 53 Resolver (lihat Firewall DNS Route 53 Resolver di Panduan Developer Amazon Route 53).

## Koneksi-koneksi yang tidak dilacak

Tidak semua aliran lalu lintas dilacak. Jika aturan grup keamanan mengizinkan aliran TCP atau UDP untuk semua lalu lintas (0.0.0. 0/0 atau: :/0) dan ada aturan yang sesuai di arah lain yang mengizinkan semua lalu lintas respons (0.0.0. 0/0 atau: :/0) untuk port apa pun (0-65535), maka aliran lalu lintas itu tidak dilacak, kecuali itu adalah bagian dari koneksi yang dilacak secara otomatis. ??? Lalu lintas tanggapan untuk aliran yang tidak dilacak akan diizinkan berdasarkan aturan-aturan ke dalam atau ke luar yang mengizinkan lalu lintas tanggapan, bukan berdasarkan informasi pelacakan.

Aliran lalu lintas yang tidak dilacak akan langsung diputus jika aturan yang memungkinkan aliran dihapus atau dimodifikasi. Misalnya, jika Anda memiliki open (0.0.0. 0/0) aturan keluar, dan Anda menghapus aturan yang memungkinkan semua (0.0.0. 0/0) lalu lintas SSH (port TCP 22) masuk ke instance (atau memodifikasinya sedemikian rupa sehingga koneksi tidak lagi diizinkan), koneksi SSH Anda yang ada ke instance segera dihentikan. Koneksi tersebut sebelumnya tidak dilacak, sehingga perubahan yang diterapkan akan memutus koneksi itu. Di sisi lain, jika Anda memiliki aturan ke dalam yang lebih sempit yang dari awal mengizinkan koneksi SSH (artinya koneksi dilacak), tetapi kemudian Anda mengubah aturan tersebut sehingga tidak lagi mengizinkan koneksi baru dari alamat

klien SSH saat ini, maka koneksi SSH yang sudah ada tidak akan terputus karena koneksi itu sudah dilacak.

## Koneksi-koneksi yang dilacak secara otomatis

Koneksi yang dibuat melalui berikut ini dilacak secara otomatis, bahkan jika konfigurasi grup keamanan tidak memerlukan pelacakan:

- Egress-only gerbang internet
- Akselerator Global Accelerator
- Gateway NAT
- Titik akhir firewall Network Firewall
- Penyeimbang Beban Jaringan
- AWS PrivateLink (antarmuka titik akhir VPC)
- AWS Lambda (Antarmuka jaringan elastis hyperplane)
- Titik akhir gateway DynamoDB — Setiap koneksi ke DynamoDB mengkonsumsi 2 entri conntrack.

## Tunjangan pelacakan koneksi

Amazon EC2 menetapkan jumlah maksimum koneksi yang dapat dilacak untuk setiap instans. Setelah jumlah maksimum tercapai, setiap paket yang dikirim atau diterima akan dihapus karena koneksi baru tidak dapat dibuat. Ketika ini terjadi, aplikasi-aplikasi yang mengirim dan menerima paket tidak akan dapat berkomunikasi dengan semestinya. Gunakan metrik performa jaringan `conntrack_allowance_available` untuk menentukan jumlah koneksi yang dilacak yang masih tersedia untuk tipe instans tersebut.

Untuk menentukan apakah paket sudah dihapus karena lalu lintas jaringan untuk instans Anda melebihi jumlah maksimum koneksi yang dapat dilacak, gunakan metrik performa jaringan `conntrack_allowance_exceeded`. Untuk informasi selengkapnya, lihat Memantau kinerja jaringan untuk pengaturan ENA pada instans EC2 Anda.

Dengan Penyeimbangan Beban Elastis, jika Anda melebihi jumlah maksimum koneksi yang dapat dilacak untuk setiap instans, kami merekomendasikan agar Anda menskalakan jumlah instans yang terdaftar dengan penyeimbang beban atau ukuran instans yang terdaftar dengan penyeimbang beban.

## Praktik terbaik pelacakan koneksi

Perutean asimetris, di mana lalu lintas masuk ke instance melalui satu antarmuka jaringan dan keluar melalui antarmuka jaringan yang berbeda, dapat mengurangi kinerja puncak yang dapat dicapai instance jika aliran dilacak.

Untuk mempertahankan kinerja puncak dan mengoptimalkan manajemen koneksi saat pelacakan koneksi diaktifkan untuk grup keamanan Anda, sebaiknya konfigurasi berikut:

- Hindari topologi routing asimetris, jika memungkinkan.
- Alih-alih menggunakan grup keamanan untuk pemfilteran, gunakan ACL jaringan.
- Jika Anda harus menggunakan grup keamanan dengan pelacakan koneksi, konfigurasikan batas waktu pelacakan koneksi idle sesingkat mungkin. Untuk detail selengkapnya tentang batas waktu pelacakan koneksi idle, lihat bagian berikut.
- Dengan batas waktu default yang lebih pendek pada instans Nitrov6, aplikasi dengan koneksi berumur panjang (seperti kumpulan koneksi database, koneksi HTTP persisten, atau beban kerja streaming) harus mengonfigurasi nilai yang sesuai `TcpEstablishedTimeout` saat peluncuran instance.
- Untuk koneksi berumur panjang, konfigurasikan TCP keep alives untuk dikirim pada interval kurang dari 5 menit untuk memastikan koneksi tetap terbuka dan mempertahankan status terlacaknya. Ini membantu mencegah koneksi terputus karena batas waktu idle dan mengurangi overhead pembentukan kembali koneksi.

Untuk informasi selengkapnya tentang penyetelan kinerja pada sistem Nitro, lihatPertimbangan sistem nitro untuk penyetelan kinerja.

## Waktu habis pelacakan koneksi idle

Grup keamanan melacak setiap koneksi yang dibuat untuk memastikan bahwa paket yang kembali dikirim seperti yang diharapkan. Ada jumlah maksimum koneksi yang dapat dilacak per instans. Koneksi yang tetap dalam keadaan idle dapat menyebabkan terbebaninya pelacakan koneksi dan menyebabkan koneksi tidak dilacak dan paket terputus. Anda sekarang dapat mengatur batas waktu untuk pelacakan koneksi pada antarmuka jaringan Elastis.

> **Note**
>
> Fitur ini hanya tersedia dengan Nitro-based instance. Anda harus menguji aplikasi Anda pada instans generasi Nitrov6 dengan waktu tunggu pelacakan koneksi default `350` kedua yang dikurangi sebelum menerapkan ke produksi.

Ada tiga batas waktu yang dapat dikonfigurasi:

- TCP menetapkan batas waktu: Batas waktu (dalam detik) untuk koneksi TCP idle dalam keadaan mapan.
  - Min: `60` detik
  - Maks: `432000` detik
  - Default: `350` detik untuk jenis instance Nitro v6 (tidak termasuk P6e-GB200); `432000` detik untuk semua jenis instance lainnya (termasuk P6e-GB200).
  - Direkomendasikan: Kurang dari `432000` detik
- Batas waktu UDP: Batas waktu (dalam detik) untuk alur UDP idle yang telah melihat lalu lintas hanya dalam satu arah atau transaksi permintaan-respons tunggal.
  - Min: `30` detik
  - Maks: `60` detik
  - Default: `30` detik
- Batas waktu aliran UDP: Batas waktu (dalam detik) untuk alur UDP idle yang diklasifikasikan sebagai alur yang telah melihat lebih dari satu transaksi permintaan-respons.
  - Min: `60` detik
  - Maks: `180` detik
  - Default: `180` detik

Anda mungkin ingin memodifikasi batas waktu default untuk salah satu kasus berikut:

- Jika Anda memantau koneksi yang dilacak menggunakan metrik performa jaringan Amazon EC2, metrik conntrack_allowance_exceeded dan conntrack_allowance_available memungkinkan Anda memantau paket yang terputus dan pemanfaatan koneksi yang dilacak untuk secara proaktif mengelola kapasitas instans EC2 dengan tindakan peningkatatan atau penurunan skala untuk membantu memenuhi permintaan koneksi jaringan sebelum memutuskan paket.

Jika Anda mengamati penurunan  conntrack_allowance_pada instans EC2 Anda, Anda dapat memperoleh manfaat dari menyetel batas waktu yang ditetapkan TCP yang lebih rendah untuk memperhitungkan sesi basi akibat klien atau kotak tengah jaringan yang tidak tepat. TCP/UDP

- Biasanya, penyeimbang beban atau firewall memiliki batas waktu idle yang Ditetapkan TCP dalam kisaran 60 hingga 90 menit. Jika Anda menjalankan beban kerja yang diharapkan akan menangani jumlah koneksi yang sangat banyak (lebih dari 100k) dari peralatan seperti firewall jaringan, Anda disarankan untuk mengonfigurasi batas waktu yang sama pada antarmuka jaringan EC2.
- Jika Anda menjalankan beban kerja yang menggunakan topologi routing asimetris, sebaiknya Anda mengonfigurasi batas waktu idle TCP Established 60 detik.
- Jika Anda menjalankan beban kerja dengan jumlah koneksi yang tinggi seperti DNS, SIP, SNMP, Syslog, Radius, dan layanan lain yang terutama menggunakan UDP untuk melayani permintaan, menyetel batas waktu 'UDP-stream' ke 60s memberikan kapasitas yang lebih tinggi scale/ performance untuk kapasitas yang ada dan untuk mencegah kegagalan abu-abu.
- Untuk TCP/UDP koneksi melalui Network Load Balancers, semua koneksi dilacak. Nilai batas waktu idle untuk aliran TCP adalah 350 detik dan aliran UDP adalah 120 detik, serta bervariasi dari nilai batas waktu tingkat antarmuka. Anda mungkin ingin mengonfigurasi batas waktu di tingkat antarmuka jaringan untuk memungkinkan lebih banyak fleksibilitas untuk batas waktu daripada default untuk penyeimbang beban.

Anda memiliki opsi untuk mengonfigurasi batas waktu pelacakan koneksi saat Anda melakukan hal berikut:

- Membuat antarmuka jaringan
- Memodifikasi atribut antarmuka jaringan
- Meluncurkan instans EC2
- Buat templat peluncuran instans EC2

## Contoh

Dalam contoh berikut, grup keamanan memiliki aturan ke dalam yang mengizinkan lalu lintas TCP dan ICMP, dan aturan ke luar yang mengizinkan semua lalu lintas ke luar.

Ke dalam

| Tipe protokol | Nomor port | Sumber |
|---|---|---|
| TCP | 22 (SSH) | 203.0.113. 1/32 |
| TCP | 80 (HTTP) | 0.0.0. 0/0 |
| TCP | 80 (HTTP) | ::/0 |
| ICMP | Semua | 0.0.0. 0/0 |

Ke luar

| Tipe protokol | Nomor port | Tujuan |
|---|---|---|
| Semua | Semua | 0.0.0. 0/0 |
| Semua | Semua | ::/0 |

Dengan koneksi jaringan langsung ke instans atau antarmuka jaringan, perilaku pelacakannya adalah sebagai berikut:

- Lalu lintas TCP masuk dan keluar pada port 22 (SSH) dilacak, karena aturan inbound memungkinkan lalu lintas dari 203.0.113. 1/32 hanya, dan tidak semua alamat IP (0.0.0. 0/0).
- Lalu lintas TCP ke dalam dan ke luar pada port 80 (HTTP) tidak akan dilacak, karena aturan ke dalam dan ke luar mengizinkan lalu lintas dari semua alamat IP.
- Lalu lintas ICMP selalu dilacak.

Jika Anda menghapus aturan ke luar untuk lalu lintas IPv4, maka semua lalu lintas IPv4 ke dalam dan ke luar akan dilacak, termasuk lalu lintas yang ada di port 80 (HTTP). Hal yang sama juga berlaku untuk lalu lintas IPv6 jika Anda menghapus aturan ke luar untuk lalu lintas IPv6.

## Aturan-aturan grup keamanan untuk kasus penggunaan yang berbeda

Anda dapat membuat grup keamanan dan menambahkan aturan-aturan yang mencerminkan peran dari instans yang dikaitkan dengan grup keamanan tersebut. Sebagai contoh, instans yang dikonfigurasi sebagai server web akan membutuhkan aturan-aturan grup keamanan

yang mengizinkan akses HTTP dan HTTPS ke dalam. Demikian juga, instans basis data akan membutuhkan aturan-aturan yang mengizinkan akses untuk jenis basis data, seperti akses melalui port 3306 untuk MySQL.

Berikut ini adalah contoh jenis aturan yang dapat Anda tambahkan ke grup keamanan untuk jenis akses tertentu.

Contoh

- Aturan-aturan server web
- Aturan-aturan server basis data
- Aturan-aturan untuk terhubung ke instans dari komputer Anda
- Aturan-aturan untuk terhubung ke instans-instans dari instans dengan grup keamanan yang sama
- Aturan untuk ping/ICMP
- Aturan-aturan server DNS
- Aturan-aturan Amazon EFS
- Aturan-aturan Penyeimbangan Beban Elastis

Lihat petunjuknya di Membuat grup keamanan dan the section called “Mengonfigurasi aturan grup keamanan”.

## Aturan-aturan server web

Aturan-aturan ke dalam berikut mengizinkan akses HTTP dan HTTPS dari alamat IP mana pun. Jika VPC Anda diaktifkan untuk IPv6, maka Anda dapat menambahkan aturan-aturan untuk mengendalikan lalu lintas HTTP dan HTTPS ke dalam dari alamat IPv6.

| Tipe protokol | Nomor protokol | Port | IP sumber | Catatan |
|---|---|---|---|---|
| TCP | 6 | 80 (HTTP) | 0.0.0. 0/0 | Mengizinkan akses HTTP ke dalam dari alamat IPv4 mana pun |
| TCP | 6 | 443 (HTTPS) | 0.0.0. 0/0 | Mengizinkan akses HTTPS ke dalam dari alamat IPv4 mana pun |

| Tipe protokol | Nomor protokol | Port | IP sumber | Catatan |
|---|---|---|---|---|
| TCP | 6 | 80 (HTTP) | ::/0 | Mengizinkan akses HTTP ke dalam dari alamat IPv6 mana pun |
| TCP | 6 | 443 (HTTPS) | ::/0 | Mengizinkan akses HTTPS ke dalam dari alamat IPv6 mana pun |

## Aturan-aturan server basis data

Aturan-aturan ke dalam berikut adalah contoh aturan yang dapat Anda tambahkan untuk akses basis data, tergantung dari jenis basis data apa yang Anda jalankan pada instans Anda. Untuk informasi selengkapnya tentang instans Amazon RDS, lihat Panduan Pengguna Amazon RDS.

Untuk IP sumber, pilih salah satu hal berikut:

- alamat IP atau rentang alamat IP tertentu (dalam notasi blok CIDR) dalam jaringan lokal Anda
- ID grup keamanan untuk sekelompok instans yang mengakses basis data

| Tipe protokol | Nomor protokol | Port | Catatan |
|---|---|---|---|
| TCP | 6 | 1433 (MS SQL) | Port default untuk mengakses basis data Microsoft SQL Server, contohnya, pada instans Amazon RDS |
| TCP | 6 | 3306 () MYSQL/ Aurora | Port default untuk mengakses basis data MySQL atau Aurora, contohnya, pada instans Amazon RDS |
| TCP | 6 | 5439 (Redshift) | Port default untuk mengakses basis data klaster Amazon Redshift. |

| Tipe protokol | Nomor protokol | Port | Catatan |
|---|---|---|---|
| TCP | 6 | 5432 (PostgreSQL) | Port default untuk mengakses basis data PostgreSQL, contohnya, pada instans Amazon RDS |
| TCP | 6 | 1521 (Oracle) | Port default untuk mengakses basis data Oracle, contohnya, pada instans Amazon RDS |

Opsional, Anda dapat membatasi lalu lintas ke luar dari server basis data Anda. Sebagai contoh, mungkin Anda ingin mengizinkan akses ke internet untuk pembaruan perangkat lunak, tetapi membatasi semua jenis lalu lintas lainnya. Anda harus terlebih dahulu menghapus aturan ke luar default yang mengizinkan semua lalu lintas ke luar.

| Tipe protokol | Nomor protokol | Port | IP Tujuan | Catatan |
|---|---|---|---|---|
| TCP | 6 | 80 (HTTP) | 0.0.0. 0/0 | Mengizinkan akses HTTP ke luar ke alamat IPv4 mana pun |
| TCP | 6 | 443 (HTTPS) | 0.0.0. 0/0 | Mengizinkan akses HTTPS ke luar ke alamat IPv4 mana pun |
| TCP | 6 | 80 (HTTP) | ::/0 | (Hanya IPv6-enabled VPC) Memungkinkan akses HTTP keluar ke alamat IPv6 apa pun |
| TCP | 6 | 443 (HTTPS) | ::/0 | (Hanya IPv6-enabled VPC) Memungkinkan akses HTTPS keluar ke alamat IPv6 apa pun |

## Aturan-aturan untuk terhubung ke instans dari komputer Anda

Untuk terhubung ke instans Anda, grup keamanan Anda harus memiliki aturan-aturan ke dalam yang mengizinkan akses SSH (untuk instans Linux) atau akses RDP (untuk instans Windows).

| Tipe protokol | Nomor protokol | Port | IP sumber |
| --- | --- | --- | --- |
| TCP | 6 | 22 (SSH) | Alamat IPv4 publik dari komputer Anda, atau rentang alamat IP di jaringan lokal Anda. Jika VPC Anda diaktifkan untuk IPv6 dan instans Anda memiliki alamat IPv6, maka Anda dapat memasukkan alamat atau rentang alamat IPv6. |
| TCP | 6 | 3389 (RDP) | Alamat IPv4 publik dari komputer Anda, atau rentang alamat IP di jaringan lokal Anda. Jika VPC Anda diaktifkan untuk IPv6 dan instans Anda memiliki alamat IPv6, maka Anda dapat memasukkan alamat atau rentang alamat IPv6. |

## Aturan-aturan untuk terhubung ke instans-instans dari instans dengan grup keamanan yang sama

Untuk mengizinkan instans yang dikaitkan dengan grup keamanan yang sama untuk saling berkomunikasi satu sama lain, Anda harus secara eksplisit menambahkan aturan untuk hal ini.

> **Note**
>
> Jika Anda mengonfigurasi rute untuk meneruskan lalu lintas antara dua instans di subnet yang berbeda melalui perangkat middlebox, Anda harus memastikan bahwa grup keamanan untuk kedua instans tersebut mengizinkan lalu lintas mengalir di antara instans. Grup keamanan untuk setiap instans harus mereferensikan alamat IP privat instans lain, atau rentang CIDR dari subnet yang berisi instans yang lain, sebagai sumbernya. Jika Anda mereferensikan grup keamanan instans lain sebagai sumbernya, hal ini tidak akan mengizinkan lalu lintas mengalir di antara instans.

Tabel berikut ini menjelaskan aturan ke dalam untuk grup keamanan yang memungkinkan instans yang dikaitkan untuk saling berkomunikasi satu sama lain. Aturan ini mengizinkan semua jenis lalu lintas.

| Tipe protokol | Nomor protokol | Port | IP sumber |
|---|---|---|---|
| -1 (Semua) | -1 (Semua) | -1 (Semua) | ID grup keamanan, atau rentang CIDR dari subnet yang berisi instans lainnya (lihat catatan). |

## Aturan untuk ping/ICMP

Perintah ping merupakan jenis lalu lintas ICMP. Untuk melakukan ping pada instans Anda, Anda harus menambahkan aturan ICMP ke dalam berikut ini.

| Tipe | Protokol | Sumber | | |
|---|---|---|---|---|
| ICMP - IPv4 Kustom | Permintaan Echo | Alamat IPv4 publik dari komputer Anda, alamat IPv4 tertentu, atau alamat IPv4 atau IPv6 dari mana saja. | | |
| Semua ICMP - IPv4 | IPv4 ICMP (1) | Alamat IPv4 publik dari komputer Anda, alamat IPv4 tertentu, atau alamat IPv4 atau IPv6 dari mana saja. | | |

Untuk menggunakan perintah ping6 untuk melakukan ping pada alamat IPv6 untuk instans Anda, Anda harus menambahkan aturan ICMPv6 ke dalam berikut ini.

| Tipe | Protokol | Sumber | | |
|---|---|---|---|---|
| Semua ICMP - IPv6 | IPv6 ICMP (58) | Alamat IPv6 dari komputer Anda, alamat IPv4 tertentu, atau alamat IPv4 atau IPv6 dari mana saja. | | |

## Aturan-aturan server DNS

Jika Anda telah mengatur instans EC2 Anda sebagai server DNS, maka Anda harus memastikan bahwa lalu lintas TCP dan UDP dapat menjangkau server DNS Anda melalui port 53.

Untuk IP sumber, pilih salah satu hal berikut:

- alamat IP atau rentang alamat IP (dalam notasi blok CIDR) di jaringan
- ID dari grup keamanan untuk serangkaian instans dalam jaringan Anda yang membutuhkan akses ke server DNS

| Tipe protokol | Nomor protokol | Port |
|---|---|---|
| TCP | 6 | 53 |
| UDP | 17 | 53 |

## Aturan-aturan Amazon EFS

Jika Anda menggunakan sistem file Amazon EFS dengan instans Amazon EC2 Anda, maka grup keamanan yang Anda kaitkan dengan target pengaitan Amazon EFS Anda harus mengizinkan lalu lintas melalui protokol NFS.

| Tipe protokol | Nomor protokol | Port | IP sumber | Catatan |
|---|---|---|---|---|
| TCP | 6 | 2049 (NFS) | ID dari grup keamanan | Mengizinkan akses NFS ke dalam dari sumber daya (termasuk target pengaitan) yang dikaitkan dengan grup keamanan ini |

Untuk mengaitkan sistem file Amazon EFS pada instans Amazon EC2 Anda, Anda harus terhubung ke instans Anda. Oleh karena itu, grup keamanan yang dikaitkan dengan instans Anda harus memiliki aturan-aturan yang mengizinkan SSH ke dalam dari komputer lokal atau jaringan lokal Anda.

| Tipe protokol | Nomor protokol | Port | IP sumber | Catatan |
|---|---|---|---|---|
| TCP | 6 | 22 (SSH) | Rentang alamat IP dari komputer lokal Anda, atau rentang alamat IP (dalam notasi blok CIDR) untuk jaringan Anda. | Mengizinkan akses SSH ke dalam dari komputer lokal Anda. |

## Aturan-aturan Penyeimbangan Beban Elastis

Jika Anda mendaftarkan instans EC2 dengan penyeimbang beban, grup keamanan yang terkait dengan penyeimbang beban Anda harus mengizinkan komunikasi dengan instans. Untuk informasi selengkapnya, lihat berikut ini dalam dokumentasi Elastic Load Balancing.

- Grup keamanan untuk Penyeimbang Beban Aplikasi
- Grup keamanan untuk Network Load Balancer
- Mengkonfigurasi grup keamanan untuk Classic Load Balancer

# NitroTPM untuk instans Amazon EC2

Nitro Trusted Platform Module (NitroTPM) adalah perangkat virtual yang disediakan oleh Sistem Nitro AWS dan sesuai dengan Spesifikasi TPM 2.0. Perangkat virtual ini akan menyimpan artefak dengan aman (seperti kata sandi, sertifikat, atau kunci enkripsi) yang digunakan untuk melakukan autentikasi terhadap instans. NitroTPM dapat membuat kunci dan menggunakan kunci tersebut untuk fungsi kriptografi (seperti melakukan hashing, penandatanganan, enkripsi, dan dekripsi).

NitroTPM menyediakan boot terukur, proses di mana bootloader dan sistem operasi membuat hash kriptografi dari setiap biner boot dan menggabungkannya dengan nilai-nilai sebelumnya di Platform Configuration Register (PCR) internal NitroTPM. Dengan boot terukur, Anda dapat memperoleh nilai PCR yang ditandatangani dari NitroTPM dan menggunakannya untuk membuktikan kepada entitas jarak jauh integritas dari perangkat lunak boot milik instans. Hal ini dikenal sebagai pengesahan jarak jauh.

Dengan NitroTPM, kunci dan rahasia dapat ditandai dengan nilai PCR tertentu sehingga kunci dan rahasia tersebut tidak akan pernah dapat diakses jika nilai PCR, dan dengan demikian integritas instansnya, berubah. Bentuk akses bersyarat khusus ini disebut sebagai sealing and unsealing. Teknologi sistem operasi, seperti BitLocker, dapat menggunakan NitroTPM untuk menyegel kunci dekripsi drive sehingga drive hanya dapat didekripsi ketika sistem operasi telah di-boot dengan benar dan dalam keadaan baik yang diketahui.

Untuk menggunakan NitroTPM, Anda harus memilih Amazon Machine Image (AMI) yang telah dikonfigurasi untuk dukungan NitroTPM, dan kemudian menggunakan AMI untuk meluncurkan instans. Nitro-based Anda dapat memilih salah satu AMI bawaan Amazon atau membuatnya sendiri.

Harga

Tidak ada biaya tambahan yang dikenakan untuk menggunakan NitroTPM. Anda hanya harus membayar untuk sumber daya dasar yang Anda gunakan.

Daftar Isi


- Persyaratan untuk menggunakan NitroTPM dengan instans Amazon EC2
- Aktifkan AMI Linux untuk NitroTPM
- Verifikasi bahwa AMI diaktifkan untuk NitroTPM
- Mengaktifkan atau menghentikan penggunaan NitroTPM pada instans Amazon EC2
- Verifikasi bahwa instans Amazon EC2 diaktifkan untuk NitroTPM

- Mengambil kunci pengesahan publik untuk instans EC2

# Persyaratan untuk menggunakan NitroTPM dengan instans Amazon EC2

Untuk meluncurkan instance dengan NitroTPM diaktifkan, Anda harus memenuhi persyaratan berikut.

Topik

- AMI
- Tipe instans
- Pertimbangan-pertimbangan

## AMI

AMI harus mengaktifkan NitroTPM.

AMI Linux

Tidak ada AMI yang telah dikonfigurasi sebelumnya. Anda harus mengkonfigurasi AMI Anda sendiri. Untuk informasi selengkapnya, lihat Aktifkan AMI Linux untuk NitroTPM.

AMI Windows

Untuk menemukan AMI AWS Windows yang telah dikonfigurasi sebelumnya untuk NitroTPM dan UEFI Secure Boot dengan kunci Microsoft, lihat M enemukan AMI Windows Server yang dikonfigurasi dengan NitroTPM dan UEFI Secure Boot di Referensi AMI Windows. AWS

> **Note**
>
> Sistem operasi - AMI harus menyertakan sistem operasi dengan driver TPM 2.0 Command Response Buffer (CRB). Sebagian besar sistem operasi saat ini menyertakan driver TPM 2.0 CRB.
> Mode boot UEFI — AMI harus dikonfigurasi untuk mode boot UEFI. Untuk informasi selengkapnya, lihat Boot Aman UEFI untuk instans Amazon EC2.

## Tipe instans

Anda harus menggunakan salah satu jenis instans tervirtualisasi berikut:

- Tujuan umum: M5, M5a, M5ad, M5d, M5dn, M5n, M5zn, M6a, M6g, M6gd, M6i, M6id, M6idn, M6in, M7a, M7g, M7gd, M7i,, M8a, M8azN, M8g, M8gb, Gd, M8gn, M8i, M8id,, M8in, M8idn, M8ine, M8ib, M8iDB, M9g, M9gd, T3, T3a, M7i-flex T4g M8i-flex
- Komputasi dioptimalkan: C5, C5a, C5ad, C5d, C5n, C6a, C6g, C6gD, C6gn, C6i, C6id, C6in, C7a, C7g, C7gD, C7gn, C7i,, C8a, C8g, C8gD, C8gD, C8gD, C8gn, C8gn, C8gD, C8gn, C8gn, C8gD, C8gn, C8gn, C8gD, C8gn, C8gn, C8gD, C8gn, C8gn, C8gD, C8gn, C8gn, C8gD, C8gn, C8gn, C8gD I, C8id,, C8in, C8in, C8iB, C9g, C9gd C7i-flex C8i-flex
- Memori dioptimalkan: R5, R5a, R5ad, R5b, R5d, R5dn, R5n, R6a, R6g, R6gd, R6i, R6id, R6idn, R6in, R7a, R7g, R7gd, R7i, R7iz, R8a, R8g, R8gd, R8gn, R8i, R8id,, R8in, R8idn, R8ib, R8iDB, R9g, R9gd,,,,,, X2idn, X2IEDn, X2IEZn, X8g, X8AEDz, X8i, z1d R8i-flex U7i-6tb U7i-8tb U7i-12tb U7in-16tb U7in-24tb U7in-32tb
- Penyimpanan dioptimalkan: D3, D3en, i3en, i4i, i7i, i7ie, i8G, i8GE, iM4GN
- Komputasi yang dipercepat: F2, G4dn, G5, G6, G6e, G6f, Gr6, Gr6f, G7, G7e, Inf1, Inf2, P5, P5e, P5en,,,, Trn2, TRN2u P6-B200 P6-B300 P6e-GB200
- High-performance komputasi: hpc6a, hpc6id, hpc8a

### Pertimbangan-pertimbangan

Pertimbangan-pertimbangan berikut ini berlaku saat Anda menggunakan NitroTPM:

- Setelah Anda meluncurkan instance menggunakan AMI dengan NitroTPM diaktifkan, jika Anda ingin mengubah jenis instans, jenis instans baru yang Anda pilih juga harus mendukung NitroTPM.
- BitLocker volume yang dienkripsi dengan NitroTPM-based kunci hanya dapat digunakan pada instance asli.
- Status NitroTPM tidak ditampilkan di konsol Amazon EC2.
- Status NitroTPM tidak disertakan dalam Snapshot Amazon EBS.
- Status NitroTPM tidak termasuk dalam gambar VM Import/Export.
- NitroTPM tidak didukung di AWS Outposts., Zona Lokal, atau Zona Panjang Gelombang.

## Aktifkan AMI Linux untuk NitroTPM

Untuk mengaktifkan NitroTPM untuk sebuah instance, Anda harus meluncurkan instance menggunakan AMI dengan NitroTPM diaktifkan. Anda harus mengkonfigurasi AMI Linux Anda dengan dukungan NitroTPM saat Anda mendaftarkannya. Anda tidak dapat mengonfigurasi dukungan NitroTPM nanti.

Untuk daftar AMI Windows yang telah dikonfigurasi sebelumnya untuk dukungan NitroTPM, lihat. Persyaratan untuk menggunakan NitroTPM dengan instans Amazon EC2

Anda harus membuat AMI dengan NitroTPM dikonfigurasi dengan menggunakan API. RegisterImage Anda tidak dapat menggunakan konsol Amazon EC2 atau VM Import/Export.

Untuk mengaktifkan AMI Linux untuk NitroTPM

1. Luncurkan instance sementara dengan AMI Linux yang Anda butuhkan. Perhatikan ID volume akarnya, yang dapat Anda temukan di konsol pada tab Penyimpanan untuk instance.
2. Setelah instance mencapai `running` status, buat snapshot dari volume root instance. Untuk informasi selengkapnya, lihat Membuat snapshot volume EBS.
3. Daftarkan snapshot yang Anda buat sebagai AMI. Dalam pemetaan perangkat blok, tentukan snapshot yang Anda buat untuk volume root.

   Berikut ini adalah contoh perintah register-image. Untuk `--tpm-support`, tentukan `v2.0`. Untuk `--boot-mode`, tentukan `uefi`.

   ```
   aws ec2 register-image \
       --name my-image \
       --boot-mode uefi \
       --architecture x86_64 \
       --root-device-name /dev/xvda \
       --block-device-mappings DeviceName=/dev/
   xvda,Ebs={SnapshotId=snap-0abcdef1234567890} \
       --tpm-support v2.0
   ```

   Berikut ini adalah contoh untuk Register-EC2Image cmdlet.

   ```
   $block = @{SnapshotId=snap-0abcdef1234567890}
   Register-EC2Image `
       -Name my-image `
       -Architecture "x86_64" `
       -RootDeviceName /dev/xvda `
       -BlockDeviceMapping @{DeviceName="/dev/xvda";Ebs=$block} `
       -BootMode Uefi `
       -TpmSupport V20
   ```
4. Hentikan instance sementara yang Anda luncurkan pada langkah 1.

# Verifikasi bahwa AMI diaktifkan untuk NitroTPM

Untuk mengaktifkan NitroTPM untuk sebuah instance, Anda harus meluncurkan instance menggunakan AMI dengan NitroTPM diaktifkan. Anda dapat menjelaskan gambar untuk memverifikasi bahwa itu diaktifkan untuk NitroTPM. Jika Anda adalah pemilik AMI, Anda dapat menjelaskan atribut `tpmSupport` gambar.

Konsol Amazon EC2 tidak ditampilkan`TpmSupport`.

AWS CLI

Untuk memverifikasi bahwa NitroTPM diaktifkan

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
    --query Images[*].TpmSupport
```

Jika NitroTPM diaktifkan untuk AMI, outputnya adalah sebagai berikut. Jika TPM tidak diaktifkan, output kosong.

```
[
    "v2.0"
]
```

Atau, jika Anda adalah pemilik AMI, Anda dapat menggunakan perintah describe-image-atribut dengan atribut. `tpmSupport`

```
aws ec2 describe-image-attribute \
    --image-id ami-0abcdef1234567890 \
    --attribute tpmSupport
```

Berikut ini adalah output contoh.

```
{
    "ImageId": "ami-0abcdef1234567890",
    "TpmSupport": {
        "Value": "v2.0"
    }
```

```
}
```

Untuk menemukan AMI dengan NitroTPM diaktifkan

Contoh berikut mencantumkan ID AMI yang Anda miliki dengan NitroTPM diaktifkan.

```
aws ec2 describe-images \
    --owners self \
    --filters Name=tpm-support,Values=v2.0 \
    --query Images[].ImageId
```

PowerShell

Untuk memverifikasi bahwa NitroTPM diaktifkan

Gunakan Get-EC2Image cmdlet.

```
Get-EC2Image `
    -ImageId ami-0abcdef1234567890 | Select TpmSupport
```

Jika NitroTPM diaktifkan untuk AMI, outputnya adalah sebagai berikut. Jika TPM tidak diaktifkan, output kosong.

```
TpmSupport
----------
v2.0
```

Atau, jika Anda adalah pemilik AMI, Anda dapat menggunakan Get-EC2ImageAttribute cmdlet dengan atribut. `tpmSupport`

```
Get-EC2ImageAttribute `
    -ImageId ami-0abcdef1234567890 `
    -Attribute tpmSupport
```

Untuk menemukan AMI dengan NitroTPM diaktifkan

Contoh berikut mencantumkan ID AMI yang Anda miliki dengan NitroTPM diaktifkan.

```
Get-EC2Image `
    -Owner self `
    -Filter @{Name="tpm-support; Values="v2.0"} | Select ImageId
```

# Mengaktifkan atau menghentikan penggunaan NitroTPM pada instans Amazon EC2

Anda dapat mengaktifkan instans Amazon EC2 untuk NitroTPM hanya saat peluncuran. Setelah instance diaktifkan untuk NitroTPM, Anda tidak dapat menonaktifkannya. Jika Anda tidak perlu lagi menggunakan NitroTPM, Anda harus mengkonfigurasi sistem operasi untuk berhenti menggunakannya.

Topik


- Meluncurkan instans dengan NitroTPM
- Menghentikan menggunakan NitroTPM pada instans


## Meluncurkan instans dengan NitroTPM

Ketika Anda meluncurkan instans dengan prasyarat, NitroTPM secara otomatis akan diaktifkan pada instans. Anda dapat mengaktifkan NitroTPM pada instance hanya saat peluncuran. Untuk informasi tentang cara meluncurkan instans, lihat Luncurkan instans Amazon EC2.

## Menghentikan menggunakan NitroTPM pada instans

Setelah meluncurkan instance dengan NitroTPM diaktifkan, Anda tidak dapat menonaktifkan NitroTPM untuk instance tersebut. Namun demikian, Anda dapat mengonfigurasi sistem operasi untuk berhenti menggunakan NitroTPM dengan menonaktifkan driver perangkat TPM 2.0 pada instans menggunakan alat-alat berikut:

- Untuk inst ans Linux, gunakan tpm-tools.
- Untuk inst ans Windows, gunakan konsol manajemen TPM (tpm.msc).

Untuk informasi selengkapnya tentang bagaimana menonaktifkan driver perangkat, lihat dokumentasi untuk sistem operasi Anda.

# Verifikasi bahwa instans Amazon EC2 diaktifkan untuk NitroTPM

Anda dapat memverifikasi apakah instans Amazon EC2 diaktifkan untuk NitroTPM. Jika dukungan NitroTPM diaktifkan pada instance, perintah kembali. `"v2.0"` Jika tidak, `TpmSupport` bidang tidak ada dalam output.

Konsol Amazon EC2 tidak menampilkan `TpmSupport` bidang.

AWS CLI

Untuk memverifikasi apakah instance diaktifkan untuk NitroTPM

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[].Instances[].TpmSupport
```

PowerShell

Untuk memverifikasi apakah instance diaktifkan untuk NitroTPM

Gunakan Get-EC2Instance cmdlet.

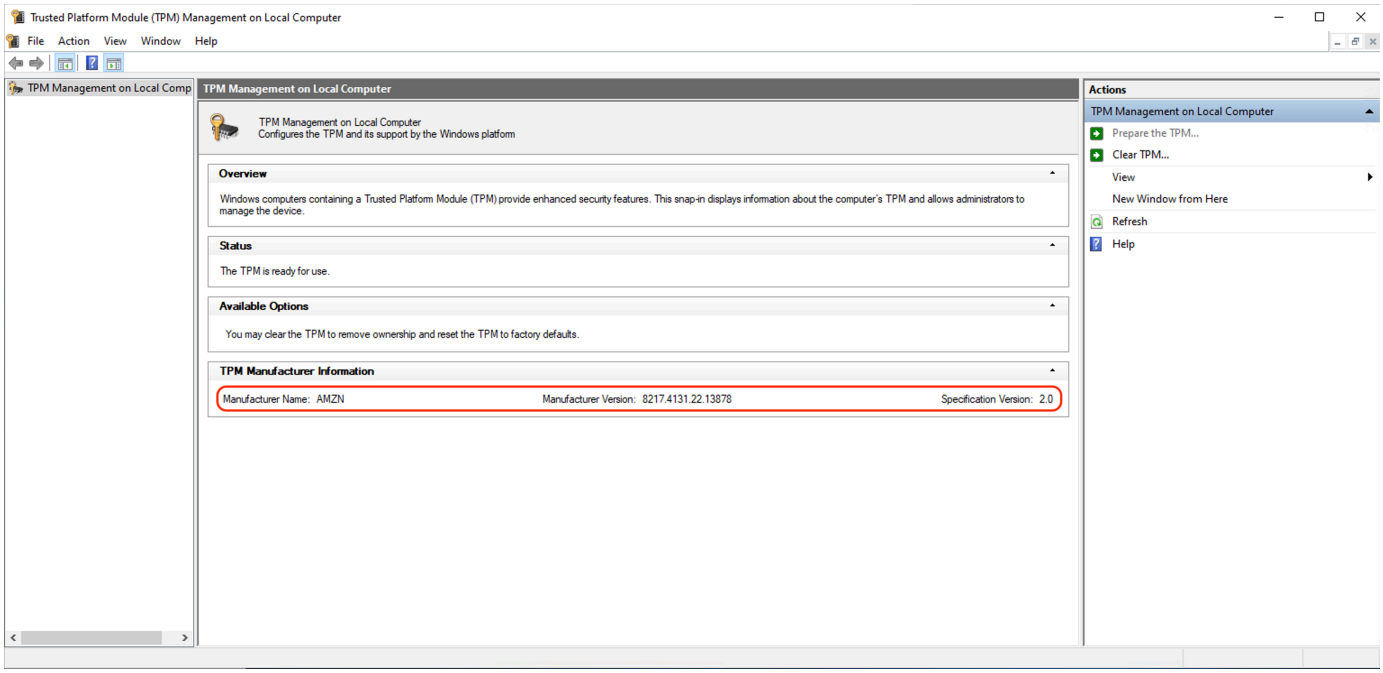
```
(Get-EC2Instance `
    -InstanceId i-1234567890abcdef0).Instances.TpmSupport
```

## Verifikasi akses NitroTPM pada instans Windows Anda

(Hanya instans Windows) Untuk memverifikasi apakah NitroTPM dapat diakses oleh Windows

1. Hubungkan ke instans Windows EC2 Anda.
2. Pada instans tersebut, jalankan program tpm.msc.

   Jendela Manajemen TPM pada Komputer Lokal akan terbuka.
3. Periksa bidang Informasi Produsen TPM. Bidang tersebut berisi nama produsen dan versi NitroTPM yang ada pada instans.




# Mengambil kunci pengesahan publik untuk instans EC2

Anda dapat mengambil kunci dukungan publik untuk instance dengan aman kapan saja.

AWS CLI

Untuk mengambil kunci dukungan publik untuk sebuah instance

Gunakan perintah get-instance-tpm-ek-pub.

Contoh 1

Contoh berikut mendapatkan kunci dukungan `rsa-2048` publik dalam `tpmt` format untuk instance yang ditentukan.

```
aws ec2 get-instance-tpm-ek-pub \
    --instance-id i-1234567890abcdef0 \
    --key-format tpmt \
    --key-type rsa-2048
```

Berikut ini adalah contoh output.

```
{
```

```
    "InstanceId": "i-01234567890abcdef",
    "KeyFormat": "tpmt",
    "KeyType": "rsa-2048",
    "KeyValue": "AAEACwADALIAIINxl2dEhLEXAMPLEUal1yT9UtduBlILZPKh2hszFGmqAAYAgABDA
    EXAMPLEAAABAOiRd7WmgtdGNoV1h/AxmW+CXExblG8pEUfNm0LOLiYnEXAMPLERqApiFa/UhvEYqN4
    Z7jKMD/usbhsQaAB1gKA5RmzuhSazHQkax7EXAMPLEzDthlS7HNGuYn5eG7qnJndRcakS+iNxT8Hvf
    0S1ZtNuItMs+Yp4SO6aU28MT/JZkOKsXIdMerY3GdWbNQz9AvYbMEXAMPLEPyHfzgVO0QTTJVGdDxh
    vxtXCOu9GYf0crbjEXAMPLEd4YTbWdDdgOKWF9fjzDytJSDhrLAOUctNzHPCd/92l5zEXAMPLEOIFA
    Ss50C0/802c17W2pMSVHvCCa9lYCiAfxH/vYKovAAE="
}
```

Contoh 2

Contoh berikut mendapatkan kunci dukungan `rsa-2048` publik dalam `der` format untuk instance yang ditentukan.

```
aws ec2 get-instance-tpm-ek-pub \
    --instance-id i-1234567890abcdef0 \
    --key-format der \
    --key-type rsa-2048
```

Berikut ini adalah contoh output.

```
{
    "InstanceId": "i-1234567890abcdef0",
    "KeyFormat": "der",
    "KeyType": "rsa-2048",
    "KeyValue": "MIIBIjANBgEXAMPLEw0BAQEFAAOCAQ8AMIIBCgKCAQEA6JF3taEXAMPLEXWH8DGZb4
    JcTFuUbykRR82bQs4uJifaKSOv5NGoEXAMPLEG8Rio3hnuMowP+6xuGxBoAHWAoDlGbO6FJrMdEXAMP
    LEnYUHvMO2GVLsc0a5ifl4buqcmd1FxqRL6I3FPwe9/REXAMPLE0yz5inhI7ppTbwxP8lmQ4qxch0x6
    tjcZ1Zs1DP0EXAMPLERUYLQ/Id/OBU7RBNMlUZ0PGG/G1cI670Zh/RytuOdx9iEXAMPLEtZ0N2A4pYX
    1+PMPK0lIOGssA5Ry03Mc8J3/3aXnOD2/ASRQ4gUBKznQLT/zTZEXAMPLEJUe8IJr2VgKIB/Ef+9gqi
    8AAQIDAQAB"
}
```

PowerShell

Untuk mengambil kunci dukungan publik untuk sebuah instance

Gunakan Get-EC2InstanceTpmEkPub cmdlet.

Contoh 1

Contoh berikut mendapatkan kunci dukungan `rsa-2048` publik dalam `tpmt` format untuk instance yang ditentukan.

```
Get-EC2InstanceTpmEkPub `
    -InstanceId i-1234567890abcdef0 `
    -KeyFormat tpmt `
    -KeyType rsa-2048
```

Contoh 2

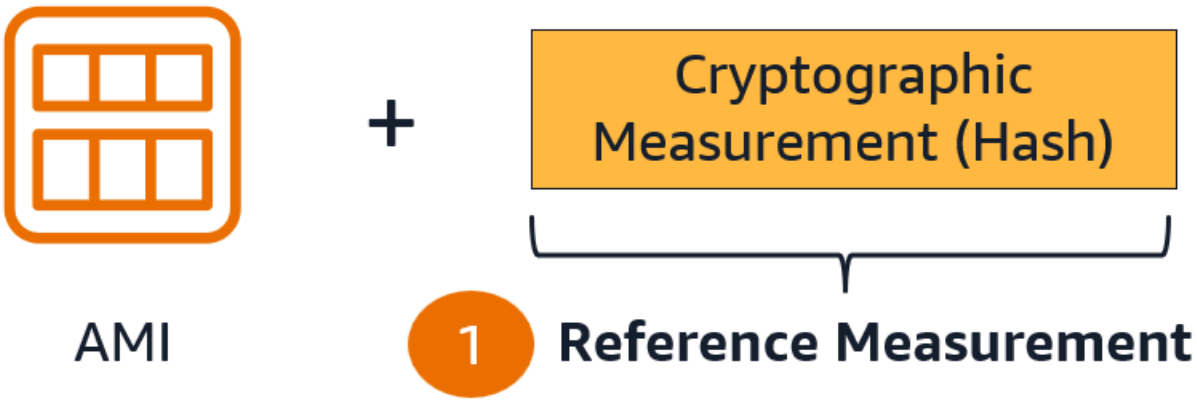
Contoh berikut mendapatkan kunci dukungan `rsa-2048` publik dalam `der` format untuk instance yang ditentukan.

```
Get-EC2InstanceTpmEkPub `
    -InstanceId i-1234567890abcdef0 `
    -KeyFormat der `
    -KeyType rsa-2048
```

# Pengesahan instans Amazon EC2

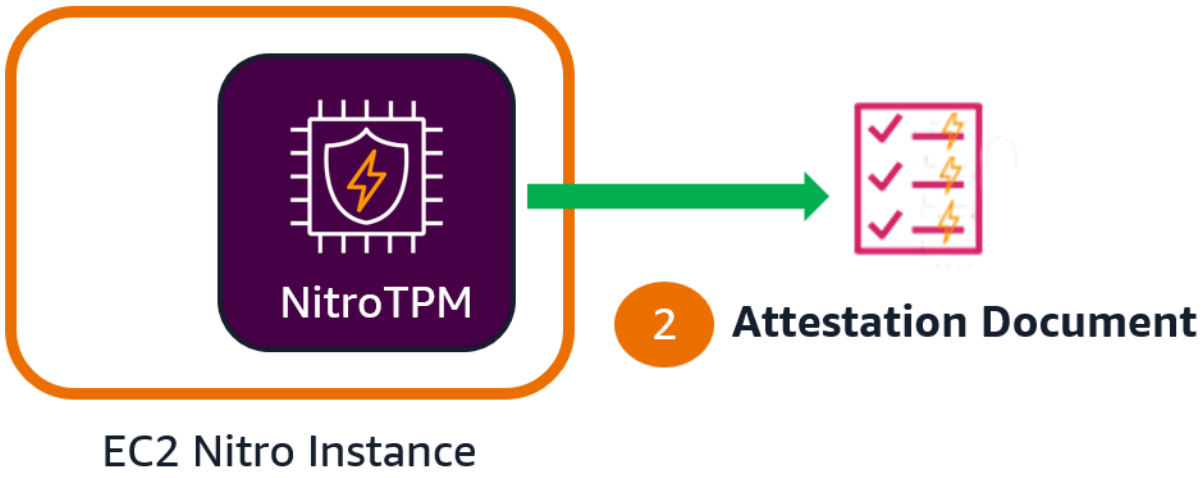
Pengesahan adalah proses yang memungkinkan Anda membuktikan secara kriptografis kepada pihak mana pun bahwa hanya perangkat lunak tepercaya, driver, dan proses boot yang berjalan pada instans Amazon EC2. Pengesahan instans Amazon EC2 didukung oleh Nitro Trusted Platform Module (NitroTPM) dan AMI yang Dapat Diatestasi.

Langkah pertama untuk pengesahan adalah membangun AMI yang Dapat Dibuktikan dan menentukan pengukuran referensi AMI tersebut. AMI yang Dapat Dibuktikan adalah AMI yang dibangun dari bawah ke atas untuk pengesahan. Pengukuran referensi adalah pengukuran semua perangkat lunak dan konfigurasi Anda yang telah Anda sertakan dalam AMI Anda. Untuk informasi lebih lanjut tentang bagaimana Anda dapat memperoleh pengukuran referensi, lihatBangun deskripsi gambar sampel.




Langkah selanjutnya adalah meluncurkan instance EC2 yang Nitro-TPM diaktifkan dengan AMI yang Dapat Diatestable. Setelah Anda meluncurkan instance, Anda dapat menggunakan alat NitroTPM untuk menghasilkan Dokumen Pengesahan. Kemudian Anda dapat membandingkan pengukuran aktual instans EC2 Anda dari Dokumen Pengesahan dengan pengukuran referensi untuk memeriksa apakah instans memiliki perangkat lunak dan konfigurasi yang Anda percayai.

Dengan membandingkan pengukuran referensi yang dihasilkan selama proses pembuatan AMI yang dapat dibuktikan dengan pengukuran yang disertakan dalam Dokumen Pengesahan instans, Anda dapat memvalidasi bahwa hanya perangkat lunak dan kode yang Anda percayai yang berjalan pada instance.



## Integrasi dengan AWS KMS

Untuk mempermudah proses membandingkan pengukuran, Anda dapat menggunakan AWS Key Management Service (AWS KMS) sebagai verifikasi untuk Dokumen Pengesahan. Dengan AWS KMS, Anda dapat membuat kebijakan kunci KMS berbasis pengesahan yang memungkinkan operasi tertentu dengan kunci KMS hanya jika Anda memberikan Dokumen Pengesahan dengan pengukuran yang sesuai dengan pengukuran referensi. Untuk melakukan ini, Anda menambahkan kunci kondisi tertentu ke kebijakan kunci KMS yang menggunakan pengukuran referensi sebagai nilai kunci kondisi, dan kemudian menentukan operasi KMS mana yang diizinkan jika kunci kondisi terpenuhi.

Saat Anda melakukan operasi KMS menggunakan kunci KMS, Anda harus melampirkan Dokumen Pengesahan ke permintaan KMS. AWS KMS kemudian memvalidasi pengukuran dari Dokumen Pengesahan terhadap pengukuran referensi dalam kebijakan kunci KMS, dan mengizinkan akses kunci hanya jika pengukuran cocok.

Selain itu, saat Anda membuat Dokumen Pengesahan untuk sebuah instance, Anda harus menentukan kunci publik untuk pasangan kunci yang Anda miliki. Kunci publik yang ditentukan termasuk dalam Dokumen Pengesahan. Ketika AWS KMS memvalidasi Dokumen Pengesahan dan memungkinkan operasi dekripsi, secara otomatis mengenkripsi respons dengan kunci publik yang disertakan dalam Dokumen Pengesahan sebelum dikembalikan. Ini memastikan bahwa respons dapat didekripsi dan digunakan hanya dengan kunci pribadi yang cocok untuk kunci publik yang disertakan dalam Dokumen Pengesahan.

Ini memastikan bahwa hanya instans yang menjalankan perangkat lunak dan kode tepercaya yang dapat melakukan operasi kriptografi menggunakan kunci KMS.

## Membuktikan lingkungan komputasi terisolasi

Secara umum, Anda dapat membangun dan mengonfigurasi instance EC2 menjadi lingkungan komputasi terisolasi, yang tidak menyediakan akses interaktif dan tidak ada mekanisme bagi administrator dan pengguna untuk mengakses data yang sedang diproses dalam instans EC2. Dengan pengesahan instans EC2, Anda dapat membuktikan kepada pihak ketiga atau layanan bahwa instans Anda berjalan sebagai lingkungan komputasi terisolasi. Untuk informasi selengkapnya, lihat Isolasi data dari operator Anda sendiri.

Sebagai contoh, lihat contoh deskripsi gambar Amazon Linux 2023 yang menciptakan lingkungan komputasi terisolasi. Anda dapat menggunakan deskripsi gambar sampel ini sebagai titik awal dan menyesuaikannya untuk memenuhi kebutuhan Anda.

# AWS model tanggung jawab bersama

NitroTPM dan AMI yang Dapat Diatestasi adalah blok bangunan yang dapat membantu Anda mengatur dan mengonfigurasi pengesahan pada instans EC2 Anda. Anda bertanggung jawab untuk mengonfigurasi AMI untuk memenuhi kasus penggunaan Anda masing-masing. Untuk informasi selengkapnya, lihat AWS Model Tanggung Jawab Bersama.

Topik


- AMI yang dapat dibuktikan
- Persiapkan AWS KMS untuk pengesahan
- Dapatkan Dokumen Pengesahan NitroTPM
- Mengintegrasikan dengan AWS KMS
- Isolasi data dari operator Anda sendiri


## AMI yang dapat dibuktikan

AMI yang dapat dibuktikan adalah Amazon Machine Image (AMI) dengan hash kriptografi yang sesuai yang mewakili semua isinya. Hash dihasilkan selama proses pembuatan AMI, dan dihitung berdasarkan seluruh konten AMI itu, termasuk aplikasi, kode, dan proses boot.

### Mempertahankan Negara yang Dapat Dibuktikan

Pengukuran instance didasarkan pada status boot awalnya. Setiap perubahan perangkat lunak atau kode yang dilakukan pada instance setelah peluncuran dan yang bertahan setelah restart akan mengubah pengukuran instance setelah restart. Jika pengukuran diubah, pengukuran tersebut menyimpang dari pengukuran referensi AMI yang dapat dibuktikan, dan instance tidak akan lagi dapat berhasil membuktikan AWS KMS setelah instance dimulai ulang. Oleh karena itu, agar AMI yang Dapat Diatestable berguna, instance harus kembali ke status boot semula setelah restart.

Selalu kembali ke status boot asli memastikan bahwa instance dapat berhasil membuktikan setelah restart. Utilitas berikut dapat digunakan untuk memastikan bahwa instans Anda tetap dapat dibuktikan setelah restart:

- `erofs`- Sistem Read-Only File yang Ditingkatkan. Utilitas ini memastikan bahwa sistem file root Anda hanya baca. Dengan utilitas ini, menulis ke sistem file, termasuk`/etc`,`/run`, dan`/var`, disimpan dalam memori dan hilang ketika instance dimulai ulang, meninggalkan sistem file root dalam keadaan peluncuran aslinya. Untuk informasi selengkapnya, lihat dokumentasi erofs.

- `dm-verity`- Memberikan perlindungan integritas untuk sistem file root read-only. Utilitas menghitung hash dari blok filesystem dan menyimpannya di baris perintah kernel. Hal ini memungkinkan kernel untuk memverifikasi integritas sistem file selama boot. Untuk informasi selengkapnya, lihat dokumentasi dm-verity.

## Persyaratan untuk membuat AMI yang dapat dibuktikan

AMI yang dapat dibuktikan memiliki persyaratan sebagai berikut:

- Sistem operasi dasar — Amazon Linux 2023 dan NixOS
- Arsitektur — `x86_64` atau `arm64` arsitektur
- Dukungan TPM — NitroTPM harus diaktifkan. Untuk informasi selengkapnya, lihat Persyaratan untuk menggunakan NitroTPM dengan instans Amazon EC2.
- Mode boot — Mode boot UEFI harus diaktifkan.

Topik

- Membuat AMI yang Dapat Dibuktikan
- Buat contoh deskripsi gambar Amazon Linux 2023
- Contoh deskripsi gambar Amazon Linux 2023
- Sesuaikan contoh deskripsi gambar Amazon Linux 2023 untuk beban kerja Anda
- Hitung pengukuran PCR untuk AMI khusus

## Membuat AMI yang Dapat Dibuktikan

Untuk membuat AMI yang Dapat Dibuktikan, Anda harus menggunakan Amazon Linux 2023 dengan KIWI Next Generation (KIWI NG). Amazon Linux 2023 menyediakan semua perangkat lunak dan utilitas yang diperlukan untuk membangun AMI yang Dapat Dibuktikan menggunakan KIWI NG.

KIWI NG adalah alat sumber terbuka untuk membangun gambar pra-konfigurasi Linux-based . KIWI NG menggunakan deskripsi gambar XML yang menentukan isi gambar. Deskripsi gambar menentukan sistem operasi dasar, perangkat lunak, konfigurasi kernel, dan skrip untuk dijalankan untuk membangun AMI siap pakai untuk kasus penggunaan tertentu.

Selama waktu pembuatan AMI, Anda perlu menggunakan `nitro-tpm-pcr-compute` utilitas untuk menghasilkan pengukuran referensi berdasarkan Unified Kernel Image (UKI) yang dihasilkan oleh

KIWI NG. Untuk informasi lebih lanjut tentang menggunakan `nitro-tpm-pcr-compute` utilitas, lihatHitung pengukuran PCR untuk AMI khusus.

AWS memberikan contoh deskripsi gambar Amazon Linux 2023 yang mencakup semua konfigurasi yang diperlukan untuk mengonfigurasi instans EC2 di lingkungan komputasi terisolasi. Untuk informasi selengkapnya, lihat Buat contoh deskripsi gambar Amazon Linux 2023.

## Buat contoh deskripsi gambar Amazon Linux 2023

AWS memberikan contoh deskripsi gambar Amazon Linux 2023 yang dapat Anda gunakan sebagai titik awal untuk membuat AMI yang Dapat Diverifikasi khusus untuk beban kerja Anda. Deskripsi gambar sampel mencakup Amazon Linux 2023 sebagai sistem operasi dasar, `dm-verity` dan `erofs` konfigurasi untuk kekekalan sistem file, dan menghapus semua akses interaktif (seperti SSH, koneksi instans EC2, dan konsol serial) untuk membuat lingkungan komputasi yang terisolasi. Untuk informasi lebih lanjut tentang deskripsi gambar sampel, lihat re po Github.

Deskripsi gambar sampel secara otomatis menginstal alat NitroTPM (`nitro-tpm-pcr-computedannitro-tpm-attest`) di gambar bawaan di `/usr/bin/` direktori. Ini memastikan bahwa alat sudah diinstal sebelumnya pada instans yang diluncurkan dari AMI.

Deskripsi gambar sampel mencakup skrip,`edit_boot_install.sh`, yang mencakup perintah yang diperlukan untuk menghasilkan pengukuran referensi. Skrip memasang file gambar disk mentah (`.raw`) yang dibuat oleh KIWI NG ke perangkat loopback, menemukan UKI, yang memiliki ekstensi `.efi` file, dan kemudian menjalankan `nitro-tpm-pcr-compute` utilitas untuk menghasilkan pengukuran referensi untuk AMI. Script secara otomatis dieksekusi oleh KIWI NG selama waktu pembuatan.

Tutorial ini menunjukkan kepada Anda bagaimana membangun deskripsi gambar sampel untuk membuat AMI yang dapat dibuktikan.

Untuk informasi selengkapnya tentang membuat deskripsi gambar Anda sendiri, lihat dokumentasi KIWI NG berikut:

- Mulai Cepat
- Deskripsi Gambar
- Contoh deskripsi gambar Amazon Linux 2023

### Prasyarat

Identitas IAM Anda harus memiliki izin berikut untuk menyelesaikan tutorial ini:

- `ebs:CompleteSnapshot`,`ebs:StartSnapshot`, dan seter `ebs:PutSnapshotBlock` usnya `arn:aws:ec2:*::snapshot/*`
- `ec2:RegisterImage` di semua sumber daya

Untuk membuat contoh deskripsi gambar Amazon Linux 2023 menggunakan KIWI NG

1. Luncurkan instans Amazon EC2 menggunakan AMI AL2023 terbaru. Untuk memastikan instans Anda memiliki ruang penyimpanan yang cukup untuk membangun AMI, pastikan Anda menyediakan setidaknya 12 GB penyimpanan.
2. Instal dependensi yang diperlukan. Perintah berikut menginstal utilitas berikut:
   - `kiwi-cli`
   - `veritysetup`
   - `erofs-utils`
   - `aws-nitro-tpm-tools`

   ```
   sudo dnf install -y kiwi-cli python3-kiwi kiwi-systemdeps-core python3-poetry-core
    qemu-img veritysetup erofs-utils git cargo aws-nitro-tpm-tools
   ```
3. Instal `coldsnap` utilitas. Utilitas ini memungkinkan Anda membuat snapshot Amazon EBS dari data gambar mentah. Anda akan menggunakan utilitas ini untuk membuat snapshot EBS dari file gambar disk mentah yang dibuat oleh KIWI NG.

   ```
   git clone https://github.com/awslabs/coldsnap.git
   cd coldsnap
   cargo install --locked coldsnap
   cd ..
   ```
4. Dapatkan contoh file deskripsi gambar.

   ```
   sudo dnf install kiwi-image-descriptions-examples
   ```

   Contoh file deskripsi gambar diunduh ke direktori berikut: `/usr/share/kiwi-image-descriptions-examples/al2023/attestable-image-example`
5. Bangun deskripsi gambar sampel menggunakan `system build` perintah KIWI NG. Perintah berikut membuat file gambar disk mentah di `./image` direktori.

```
sudo kiwi-ng \
--color-output \
--loglevel 0 \
system build \
--description /usr/share/kiwi-image-descriptions-examples/al2023/attestable-image-example \
--target-dir ./image
```

Untuk informasi selengkapnya, lihat dokumentasi pembuatan sistem kiwi-ng.

6. Dapatkan pengukuran referensi untuk AMI. Pengukuran dihasilkan oleh `nitro-tpm-pcr-compute` utilitas selama waktu pembuatan gambar pada langkah sebelumnya. Anda dapat menemukan pengukuran referensi di file berikut: `./image/pcr_measurements.json`.

   Pengukuran disediakan dalam format JSON berikut:

   ```
   {
     "Measurements": {
       "HashAlgorithm": "SHA384 { ... }",
       "PCR4": "PCR4_measurement",
       "PCR7": "PCR7_measurement",
       "PCR12": "PCR12_measurement"
     }
   }
   ```

7. Gunakan `coldsnap` utilitas untuk mengunggah gambar disk mentah yang dibuat oleh KIWI NG ke snapshot EBS. Perintah mengembalikan ID snapshot. Catat ID, Anda akan membutuhkannya untuk langkah berikutnya.

   ```
   SNAPSHOT=$(.cargo/bin/coldsnap upload ./image/al2023*.raw)
   echo "Created snapshot: $SNAPSHOT"
   ```

   Untuk informasi selengkapnya tentang `coldsnap` utilitas, lihat repo coldsnap GitHub .

8. Daftarkan AMI berkemampuan TPM 2.0 dengan mode boot UEFI menggunakan snapshot dari langkah sebelumnya. Untuk`--architecture`, tentukan `x86_64` untuk Intel, atau `arm64` untuk Graviton.

   ```
   aws ec2 register-image \
   --name "attestable_isolated_al2023_ami" \
   --virtualization-type hvm \
   ```

```
--boot-mode uefi \
--architecture x86_64|arm64 \
--root-device-name /dev/xvda \
--block-device-mappings DeviceName=/dev/xvda,Ebs={SnapshotId=${SNAPSHOT}} \
--tpm-support v2.0 \
--ena-support
```

## Contoh deskripsi gambar Amazon Linux 2023

Contoh deskripsi gambar Amazon Linux 2023 memiliki karakteristik sebagai berikut:

1. Boot Unified Kernel Image (UKI) — Boot menggunakan biner tunggal yang ditandatangani yang menggabungkan parameter kernel,`initrd`, dan boot menjadi satu gambar yang tidak dapat diubah.
2. Read-only sistem file root — Gunakan Enhanced Read-Only File System (`erofs`) dengan perlindungan dm-verity untuk memastikan bahwa sistem file root tidak dapat dimodifikasi dan mempertahankan verifikasi integritas kriptografi.
3. Sistem file overlay sementara - Buat sistem file overlay sementara yang memungkinkan penulisan sementara ke direktori seperti`/etc`,, dan. `/run` `/var` Karena sistem file overlay ini hanya ada di memori, semua perubahan secara otomatis hilang ketika instance reboot, memastikan sistem kembali ke keadaan tepercaya semula.
4. Metode akses jarak jauh yang dinonaktifkan — Hapus mekanisme akses jarak jauh berikut untuk mencegah akses jarak jauh:

| Metode Akses | Deskripsi | Implementasi deskripsi gambar |
|---|---|---|
| SSH | Tidak termasuk server OpenSSH. Membuat instance secara inheren tidak mampu menangani lalu lintas SSH. | Abaikan `openssh-server` paket* |
| Data Pengguna | Menghapus Cloud-init. Menghilangkan kemampuan operator untuk menyediakan data pengguna ke instance dan menjalankan skrip waktu boot-time. | Abaikan `cloud-init-cfg-ec2` |

| Metode Akses | Deskripsi | Implementasi deskripsi gambar |
| --- | --- | --- |
| | | paket `cloud-init` dan * |
| Chrony | Menonaktifkan port perintah chrony. Mencegah operator menjalankan perintah chrony pada instans yang sedang berjalan. | Abaikan `amazon-chrony-config` paket* |
| MOTD | Menghapus paket MOTD. Menghilangkan kemampuan operator untuk mengubah pesan atau fungsionalitas pada instans yang sedang berjalan. | Abaikan `update-motd` paket* |
| AWS SSM | Menghapus agen AWS SSM. Mencegah akses jarak jauh ke instans yang sedang berjalan menggunakan AWS SSM. | Abaikan `amazon-ssm-agent` paket* |
| EC2 Instance Connect | Menghapus paket EC2 Instance Connect. Menonaktifkan akses SSH menggunakan alat ini. | Abaikan `ec2-instance-connect` paket* |
| Konsol Serial | Menonaktifkan konsol serial. Memastikan bahwa akses konsol tidak tersedia untuk menjalankan instance dan menghapus kemampuan operator untuk masuk ke konsol serial. | Dinonaktifkan melalui parameter baris perintah kernel |

* Untuk informasi lebih lanjut, lihat Elem en Deskripsi Gambar.

## Sesuaikan contoh deskripsi gambar Amazon Linux 2023 untuk beban kerja Anda

Anda dapat menyesuaikan contoh deskripsi gambar Amazon Linux 2023 dan menyertakan paket perangkat lunak, skrip, dan file yang diperlukan untuk beban kerja spesifik Anda. Kustomisasi dicapai dengan menambahkan atau memodifikasi berbagai elemen dalam deskripsi gambar KIWI NG.

Topik


- Manajemen repositori
- Manajemen paket
- Menambahkan file dan direktori
- Menambahkan skrip kustom


### Manajemen repositori

Secara default, deskripsi gambar sampel menyertakan satu `<repository>` elemen yang menunjuk ke titik akhir cermin untuk repositori inti Amazon Linux 2023. Jika diperlukan, Anda dapat menambahkan referensi ke repositori lain untuk menginstal perangkat lunak yang Anda butuhkan.

Deskripsi gambar sampel menggunakan manajer `dnf` paket, seperti yang didefinisikan dalam `<packagemanager>` elemen.

Untuk informasi selengkapnya tentang menambahkan repositori, lihat Men yiapkan Repositori.

### Manajemen paket

Secara default, deskripsi gambar sampel mencakup semua paket yang diperlukan untuk membuat AMI yang dapat dibuktikan Amazon Linux 2023 untuk lingkungan komputasi terisolasi dengan sistem file `erofs` read-only.

Anda dapat menyertakan paket perangkat lunak tambahan dalam deskripsi gambar dengan menambahkannya ke `<packages>` elemen dalam deskripsi gambar. `<packages>`Elemen mendefinisikan semua perangkat lunak yang harus diinstal ke AMI.

Anda juga dapat menggunakan `<packages>` elemen untuk menghapus atau menghapus paket perangkat lunak tertentu.

Untuk informasi selengkapnya tentang menambahkan atau menghapus paket dalam deskripsi gambar, lihat Men ambahkan dan Menghapus Paket.

### Menambahkan file dan direktori

Deskripsi gambar sampel mencakup direktori pohon overlay (`/root/`). Direktori pohon overlay adalah direktori yang berisi file dan direktori yang akan disalin ke dalam gambar selama proses pembuatan gambar. Setiap file dan direktori yang Anda tempatkan ke direktori pohon overlay akan disalin langsung ke sistem file root gambar selama proses pembuatan gambar.

Direktori pohon overlay disalin ke dalam gambar setelah semua paket diinstal. File baru ditambahkan dan file yang ada diganti.

Menambahkan skrip kustom

Deskripsi gambar sampel mencakup satu skrip khusus,`edit_boot_install.sh`. Skrip ini mencakup perintah yang diperlukan untuk menjalankan `nitro-tpm-pcr-compute` utilitas, yang menghasilkan pengukuran referensi berdasarkan konten gambar. Script ini dipanggil segera setelah bootloader diinstal.

Jika diperlukan, Anda dapat menyertakan skrip kustom Anda sendiri dalam deskripsi gambar untuk melakukan tugas atau konfigurasi selama proses pembuatan gambar atau pada boot pertama gambar. Menggunakan skrip memungkinkan Anda untuk menyesuaikan gambar Anda dengan cara yang tidak dapat dicapai dengan menggunakan deskripsi gambar saja.

Untuk menyertakan skrip kustom dalam deskripsi gambar Anda, Anda perlu menamainya dengan benar berdasarkan jenis skrip, dan menambahkannya ke direktori yang sama dengan `appliance.kiwi` file. KIWI NG secara otomatis mendeteksi dan mengeksekusi skrip jika mereka diberi nama dengan benar dan ditempatkan di lokasi yang benar, tanpa perlu mereferensikan secara eksplisit dalam file deskripsi gambar.

Untuk informasi selengkapnya tentang skrip yang didukung oleh KIWI NG, lihat User-Defined Skrip.

## Hitung pengukuran PCR untuk AMI khusus

`nitro-tpm-pcr-compute`Utilitas ini memungkinkan Anda menghasilkan pengukuran referensi untuk AMI yang Dapat Dibuktikan selama waktu pembuatan berdasarkan Unified Kernel Image (UKI) -nya.

Contoh deskripsi gambar Amazon Linux 2023 secara otomatis menginstal utilitas di gambar bawaan di `/usr/bin/` direktori. Deskripsi gambar sampel juga menyertakan skrip dengan perintah yang diperlukan untuk menjalankan utilitas untuk menghasilkan pengukuran referensi selama waktu pembuatan gambar. Jika Anda menggunakan deskripsi gambar sampel, Anda tidak perlu menginstal utilitas atau menjalankannya secara manual. Untuk informasi selengkapnya, lihat Buat contoh deskripsi gambar Amazon Linux 2023.

Instal utilitas **`nitro-tpm-pcr-compute`**

Jika Anda menggunakan Amazon Linux 2023, Anda dapat menginstal `nitro-tpm-pcr-compute` utilitas dari repositori Amazon Linux sebagai berikut.

```
sudo yum install aws-nitro-tpm-tools
```

Alat diinstal di `/usr/bin` direktori.

Gunakan utilitas **`nitro-tpm-pcr-compute`**

Utilitas menyediakan satu perintah,`nitro-tpm-pcr-compute`, untuk menghasilkan pengukuran referensi.

Saat Anda menjalankan perintah, Anda harus menentukan yang berikut:

- Unified kernel image (`UKI.efi`) - Diperlukan untuk boot Standar dan UEFI.

Untuk menghasilkan pengukuran referensi untuk AMI yang Dapat Dibuktikan:

Gunakan perintah dan parameter berikut:

```
/usr/bin/nitro-tpm-pcr-compute \
--image UKI.efi
```

Utilitas mengembalikan pengukuran referensi dalam format JSON berikut:

```
{
  "Measurements": {
    "HashAlgorithm": "SHA384 { ... }",
    "PCR4": "PCR4_measurement",
    "PCR7": "PCR7_measurement",
    "PCR12": "PCR12_measurement"
  }
}
```

Untuk contoh praktis tentang cara menggunakan `nitro-tpm-pcr-compute` utilitas, lihat `edit_boot_install.sh` skrip yang disertakan dalam contoh deskripsi gambar Amazon Linux 2023.

# Persiapkan AWS KMS untuk pengesahan

> **Note**
>
> Jika Anda membuktikan layanan pihak ketiga, Anda harus membangun mekanisme kustom Anda sendiri untuk menerima, mengurai, dan memvalidasi Dokumen Pengesahan. Untuk informasi selengkapnya, lihat Validasi Dokumen Pengesahan NitroTPM.

Setelah membuat AMI yang Dapat Diatestasi, Anda harus memiliki pengukuran referensi yang dapat Anda gunakan untuk memvalidasi permintaan dari instans Amazon EC2. AWS KMS menyediakan dukungan bawaan untuk pengesahan dengan NitroTPM.

Untuk AWS KMS kunci yang Anda gunakan untuk mengenkripsi data rahasia, tambahkan kebijakan kunci yang mengizinkan akses kunci hanya jika permintaan API menyertakan Dokumen Pengesahan dengan pengukuran yang sesuai dengan pengukuran referensi yang Anda buat selama proses pembuatan AMI yang dapat dibuktikan. Gunakan pengukuran PCR4 dan PCR12 untuk boot standar atau pengukuran PCR7 untuk Boot Aman. Ini memastikan bahwa hanya permintaan dari instance yang diluncurkan menggunakan AMI yang dapat dikonfirmasi yang dapat melakukan operasi kriptografi menggunakan kunci. AWS KMS

AWS KMS menyediakan`kms:RecipientAttestation:NitroTPMPCR4`,`kms:RecipientAttestation:NitroTPM` dan kunci `kms:RecipientAttestation:NitroTPMPCR12` kondisi yang memungkinkan Anda membuat kondisi berbasis pengesahan untuk kebijakan kunci NitroTPM KMS. Untuk informasi selengkapnya, lihat K unci kondisi untuk NitroTPM.

Misalnya, kebijakan AWS KMS kunci berikut mengizinkan akses kunci hanya jika permintaan berasal dari instance dengan profil instans terlampir, dan jika permintaan menyertakan Dokumen Pengesahan dengan nilai PCR 4 dan PCR 12 tertentu. `MyEC2InstanceRole`

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow requests from instances with attested AMI only",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/MyEC2InstanceRole"
      },
```

```
      "Action": [
        "kms:Decrypt",
        "kms:GenerateDataKey",
        "kms:GenerateRandom"
      ],
      "Resource": "*",
      "Condition": {
        "StringEqualsIgnoreCase": {

 "kms:RecipientAttestation:NitroTPMPCR4":"EXAMPLE6b9b3d89a53b13f5dfd14a1049ec0b80a9ae4b159adde4

 "kms:RecipientAttestation:NitroTPMPCR12":"000000000000000000000000000000000000000000000000000
        }
      }
    }
  ]
}
```

## Dapatkan Dokumen Pengesahan NitroTPM

Dokumen Pengesahan adalah komponen kunci dari proses pengesahan NitroTPM. Ini berisi serangkaian pengukuran kriptografi yang dapat digunakan untuk memverifikasi identitas instance dan untuk membuktikan bahwa itu hanya menjalankan perangkat lunak tepercaya. Anda dapat menggunakan Dokumen Pengesahan dengan AWS KMS, yang menyediakan dukungan bawaan untuk pengesahan NitroTPM, atau untuk membangun mekanisme pengesahan kriptografi Anda sendiri.

`nitro-tpm-attest`Utilitas ini memungkinkan Anda mengambil Dokumen Pengesahan NitroTPM yang ditandatangani untuk instans Amazon EC2 selama runtime.

Contoh deskripsi gambar Amazon Linux 2023 secara otomatis menginstal utilitas di gambar bawaan di `/usr/bin/` direktori. Ini memastikan bahwa utilitas sudah diinstal sebelumnya pada instans yang diluncurkan menggunakan AMI. Anda tidak perlu menginstal utilitas secara manual. Untuk informasi selengkapnya, lihat Buat contoh deskripsi gambar Amazon Linux 2023.

Topik

- Instal utilitas nitro-tpm-attest
- Gunakan utilitas nitro-tpm-attest
- Isi Dokumen Pengesahan NitroTPM
- Validasi Dokumen Pengesahan NitroTPM

## Instal utilitas **`nitro-tpm-attest`**

Jika Anda menggunakan Amazon Linux 2023, Anda dapat menginstal `nitro-tpm-attest` utilitas dari repositori Amazon Linux sebagai berikut.

```
sudo yum install aws-nitro-tpm-tools
```

## Gunakan utilitas **`nitro-tpm-attest`**

Utilitas menyediakan satu perintah,`nitro-tpm-attest`, untuk mengambil Dokumen Pengesahan. Perintah mengembalikan Dokumen Pengesahan yang dikodekan dalam Concise Binary Object Representation (CBOR) dan ditandatangani menggunakan CBOR Object Signing and Encryption (COSE).

Saat Anda menjalankan perintah, Anda dapat menentukan parameter opsional berikut:

- `public-key`Kunci publik yang dapat digunakan oleh AWS KMS atau layanan eksternal untuk mengenkripsi data respons sebelum dikembalikan. Ini memastikan bahwa hanya penerima yang dituju, yang memiliki kunci pribadi, yang dapat mendekripsi data. Misalnya, jika Anda melakukan pengesahan dengan AWS KMS, layanan mengenkripsi data plaintext dengan kunci publik di Dokumen Pengesahan, dan mengembalikan ciphertext yang dihasilkan di bidang dalam respons. `CiphertextForRecipient` Hanya kunci RSA yang didukung.
- `user-data`— Data pengguna dapat digunakan untuk mengirimkan data tambahan yang ditandatangani ke layanan eksternal. Data pengguna ini dapat digunakan untuk menyelesaikan protokol yang disepakati antara instance yang meminta dan layanan eksternal. Tidak digunakan untuk pengesahan dengan. AWS KMS
- `nonce`— Nonce dapat digunakan untuk mengatur otentikasi tantangan respon antara instance dan layanan eksternal untuk membantu mencegah serangan peniruan identitas. Menggunakan nonce memungkinkan layanan eksternal untuk memverifikasi bahwa ia berinteraksi dengan instans langsung dan bukan peniru yang menggunakan kembali Dokumen Pengesahan lama. Tidak digunakan untuk pengesahan dengan. AWS KMS

Untuk mengambil Dokumen Pengesahan

Gunakan perintah berikut dan parameter opsional:

```
/usr/bin/nitro-tpm-attest \
--public-key rsa_public_key \
```

```
--user-data user_data \
--nonce nonce
```

Untuk contoh lengkap yang menunjukkan cara menghasilkan pasangan kunci RSA, dan cara meminta pengesahan dengan kunci publik, lihat repo nitro-tpm-atest. GitHub

## Isi Dokumen Pengesahan NitroTPM

Dokumen Pengesahan dihasilkan oleh NitroTPM dan ditandatangani oleh Nitro Hypervisor. Ini mencakup serangkaian nilai register konfigurasi platform (PCR) yang terkait dengan instans Amazon EC2. PCR berikut disertakan dalam Dokumen Pengesahan:

> **Important**
>
> PCR0 dan PCR1 umumnya digunakan untuk mengukur kode boot awal, yang dikendalikan oleh. AWS Untuk memungkinkan pembaruan aman kode boot awal, PCR ini akan selalu berisi nilai konstan.

- `PCR0`- Kode yang Dapat Dieksekusi Firmware Sistem Inti
- `PCR1`- Data Firmware Sistem Inti
- `PCR2`- Kode yang dapat dieksekusi yang diperpanjang atau dapat dicolokkan
- `PCR3`- Data Firmware yang Diperpanjang atau Dapat Dicolokkan
- `PCR4`— Kode Boot Manager
- `PCR5`- Konfigurasi Kode Boot Manager dan Data dan Tabel Partisi GPT
- `PCR6`— Spesifikasi Produsen Platform Host
- `PCR7`— Kebijakan Boot Aman
- `PCR8 - 15`- Ditetapkan untuk digunakan oleh Sistem Operasi Statis
- `PCR16`— Men-debug
- `PCR23`— Dukungan Aplikasi

PCR4, PCR7, dan PCR12 secara khusus digunakan untuk memvalidasi bahwa instance diluncurkan menggunakan AMI yang dapat dibuktikan. PCR4 dan PCR12 dapat digunakan untuk memvalidasi dengan boot standar, dan PCR7 dapat digunakan untuk memvalidasi dengan Secure Boot.

- PCR4 (Boot Manager Code) — Ketika sebuah instance dimulai, NitroTPM membuat hash kriptografi dari semua binari yang dieksekusi oleh lingkungan UEFI-nya. Dengan AMI yang dapat dibuktikan, binari boot ini menyematkan hash yang mencegah pemuatan binari di masa mendatang yang tidak memiliki hash yang cocok. Dengan cara ini, hash biner boot tunggal dapat menggambarkan dengan tepat kode apa yang akan dijalankan sebuah instance.
- PCR7 (Kebijakan Boot Aman) - Biner boot UEFI dapat ditandatangani dengan kunci penandatanganan Boot Aman UEFI. Ketika UEFI Secure Boot diaktifkan, UEFI akan mencegah eksekusi binari boot UEFI yang tidak sesuai dengan kebijakan yang dikonfigurasi. PCR7 berisi hash kebijakan Boot Aman UEFI instance.

  Jika Anda perlu mempertahankan kebijakan KMS tunggal yang tetap berlaku di seluruh pembaruan instans, Anda dapat membuat kebijakan yang memvalidasi terhadap PCR7 untuk memvalidasi sertifikat Boot Aman UEFI. Selama pembuatan AMI yang Dapat Diatestasi, Anda kemudian dapat menandatangani biner boot dengan sertifikat Anda dan menginstalnya sebagai satu-satunya sertifikat yang diizinkan di AMI. UEFI-data Perlu diingat bahwa model ini mengharuskan Anda untuk tetap membuat sertifikat baru, menginstalnya di kebijakan, dan memperbarui AMI jika Anda ingin mencegah instans yang diluncurkan dari AMI lama (tidak tepercaya) meneruskan kebijakan KMS Anda.
- PCR12 - Berisi hash dari baris perintah yang diteruskan ke biner boot UEFI. Diperlukan bersama dengan PCR4 untuk boot standar untuk memvalidasi baris perintah tidak dimodifikasi.

## Validasi Dokumen Pengesahan NitroTPM

> **Note**
>
> Topik ini ditujukan untuk pengguna yang menggunakan layanan manajemen kunci pihak ketiga, dan perlu membangun mekanisme validasi Dokumen Pengesahan mereka sendiri.

Topik ini memberikan gambaran rinci tentang seluruh alur pengesahan NitroTPM. Ini juga membahas apa yang dihasilkan oleh sistem AWS Nitro ketika Dokumen Pengesahan diminta, dan menjelaskan bagaimana layanan manajemen kunci harus memproses Dokumen Pengesahan.

Topik


- Dokumen Pengesahan
- Validasi Dokumen Pengesahan

Tujuan pengesahan adalah untuk membuktikan bahwa sebuah instance adalah entitas yang dapat dipercaya, berdasarkan kode dan konfigurasi yang sedang dijalankan. Akar kepercayaan untuk contoh berada di dalam sistem AWS Nitro, yang menyediakan Dokumen Pengesahan.

Dokumen Pengesahan ditandatangani oleh AWS Nitro Attestation Public Key Infrastructure (PKI), yang mencakup otoritas sertifikat yang diterbitkan yang dapat dimasukkan ke dalam layanan apa pun.

### Dokumen Pengesahan

Dokumen Pengesahan dikodekan dalam Representasi Objek Biner Ringkas (CBOR), dan ditandatangani menggunakan CBOR Object Signing and Encryption (COSE).

Untuk informasi lebih lanjut tentang CBOR, lihat RFC 8949: Representasi Objek Biner Ringkas (CBOR).

### Spesifikasi Dokumen Pengesahan

Berikut ini menunjukkan struktur Dokumen Pengesahan.

```
AttestationDocument = {
    module_id: text,                    ; issuing Nitro hypervisor module ID
    timestamp: uint .size 8,            ; UTC time when document was created, in
                                        ; milliseconds since UNIX epoch
    digest: digest,                     ; the digest function used for calculating the
                                        ; register values
    nitrotpm_pcrs: { + index => pcr },  ; map of PCRs at the moment the Attestation
 Document was generated
    certificate: cert,                  ; the public key certificate for the public
 key
                                        ; that was used to sign the Attestation
 Document
    cabundle: [* cert],                 ; issuing CA bundle for infrastructure
 certificate
    ? public_key: user_data,            ; an optional DER-encoded key the attestation
                                        ; consumer can use to encrypt data with
    ? user_data: user_data,             ; additional signed user data, defined by
 protocol
    ? nonce: user_data,                 ; an optional cryptographic nonce provided by
 the
                                        ; attestation consumer as a proof of
 authenticity
}
```

```
cert = bytes .size (1..1024)       ; DER encoded certificate
user_data = bytes .size (0..1024)
pcr = bytes .size (32/48/64)       ; PCR content
index = 0..31
digest = "SHA384"
```

Parameter opsional dalam Dokumen Pengesahan (`public_key`,`user_data`, dannonce) dapat digunakan untuk membuat protokol validasi khusus antara instance pengesahan dan layanan eksternal.

Validasi Dokumen Pengesahan

Saat Anda meminta Dokumen Pengesahan dari Nitro Hypervisor, Anda menerima gumpalan biner yang berisi Dokumen Pengesahan yang ditandatangani. Dokumen Pengesahan yang ditandatangani adalah objek CBOR-encoded, COSE-signed (menggunakan struktur tanda tangan Cose_sign1). Proses validasi keseluruhan meliputi langkah-langkah berikut:

1. Dekode objek CBOR dan memetakannya ke struktur Cose_sign1.
2. Ekstrak Dokumen Pengesahan dari struktur Cose_sign1.
3. Verifikasi rantai sertifikat.
4. Pastikan bahwa Dokumen Pengesahan ditandatangani dengan benar.

Dokumen Pengesahan ditandatangani oleh AWS Nitro Attestation PKI, yang mencakup sertifikat root untuk partisi komersial. AWS Sertifikat root dapat diunduh dari https://aws-nitro-enclaves.amazonaws.com/AWS_NitroEnclaves_Root-G1.zip, dan dapat diverifikasi menggunakan sidik jari berikut.

```
64:1A:03:21:A3:E2:44:EF:E4:56:46:31:95:D6:06:31:7E:D7:CD:CC:3C:17:56:E0:98:93:F3:C6:8F:79:BB:5B
```

Sertifikat root didasarkan pada kunci AWS Certificate Manager pribadi Otoritas Sertifikat Pribadi (AWS Private CA) dan memiliki masa pakai 30 tahun. Subjek PCA memiliki format berikut.

```
CN=aws.nitro-enclaves, C=US, O=Amazon, OU=AWS
```

Topik

- COSE dan CBOR
- Validitas semantik

- Validitas sertifikat
- Validitas rantai sertifikat

COSE dan CBOR

Biasanya, struktur tanda tangan Cose_sign1 digunakan ketika hanya satu tanda tangan yang akan ditempatkan pada pesan. Parameter yang berhubungan dengan konten dan tanda tangan ditempatkan di header yang dilindungi daripada memiliki pemisahan Cose_sign. Struktur dapat dikodekan sebagai tag atau tidak ditandai, tergantung pada konteks yang akan digunakan. Struktur Cose_sign1 yang ditandai diidentifikasi oleh tag CBOR 18.

Objek CBOR yang membawa tubuh, tanda tangan, dan informasi tentang tubuh dan tanda tangan disebut struktur Cose_sign1. Struktur Cose_sign1 adalah array CBOR. Array mencakup bidang berikut.

```
[
  protected:   Header,
  unprotected: Header,
  payload:     This field contains the serialized content to be signed,
  signature:   This field contains the computed signature value.
]
```

Dalam konteks Dokumen Pengesahan, array mencakup yang berikut ini.

```
18(/* COSE_Sign1 CBOR tag is 18 */
    {1: -35}, /* This is equivalent with {algorithm: ECDS 384} */
    {}, /* We have nothing in unprotected */
    $ATTESTATION_DOCUMENT_CONTENT /* Attestation Document */,
    signature /* This is the signature */
)
```

Untuk informasi lebih lanjut tentang CBOR, lihat RFC 8949: Representasi Objek Biner Ringkas (CBOR).

Validitas semantik

Dokumen Pengesahan akan selalu memiliki bundel CA mereka dalam urutan berikut.

```
[ ROOT_CERT - INTERM_1 - INTERM_2 .... - INTERM_N]
      0          1          2             N - 1
```

Ingatlah urutan ini, karena beberapa alat yang ada, seperti Java CertPath dari Java PKI API Programmer Guide, mungkin mengharuskannya dipesan secara berbeda.

Untuk memvalidasi sertifikat, mulai dari bundel CA Dokumen Pengesahan dan buat rantai yang diperlukan, Dim `TARGET_CERT` ana sertifikat dalam Dokumen Pengesahan.

```
[TARGET_CERT, INTERM_N, ..... , INTERM_2, INTERM_1, ROOT_CERT]
```

Validitas sertifikat

Untuk semua sertifikat dalam rantai, Anda harus memastikan bahwa tanggal saat ini berada dalam periode validitas yang ditentukan dalam sertifikat.

Validitas rantai sertifikat

Secara umum, rantai beberapa sertifikat mungkin diperlukan, yang terdiri dari sertifikat pemilik kunci publik yang ditandatangani oleh satu CA, dan nol atau lebih sertifikat CA tambahan yang ditandatangani oleh CA lain. Rantai-rantai semacam itu, yang disebut jalur sertifikasi, diperlukan karena pengguna kunci publik hanya diinisialisasi dengan sejumlah kunci publik CA terjamin. Prosedur validasi jalur sertifikasi untuk PKI internet didasarkan pada algoritma yang disediakan. X.509 Pemrosesan jalur sertifikasi memverifikasi pengikatan antara nama subjek yang dibedakan nama and/or subjek nama alternatif dan kunci publik subjek. Pengikatan dibatasi oleh batasan yang ditentukan dalam sertifikat yang terdiri dari jalur dan input yang ditentukan oleh pihak yang mengandalkan. Kendala dasar dan ekstensi kendala kebijakan memungkinkan logika pemrosesan jalur sertifikasi untuk mengotomatiskan proses pengambilan keputusan.

> **Note**
>
> CRL harus dinonaktifkan saat melakukan validasi.

Menggunakan Java, mulai dari jalur root dan rantai sertifikat yang dihasilkan, validasi rantai adalah sebagai berikut.

```
validateCertsPath(certChain, rootCertficate) {
    /* The trust anchor is the root CA to trust */
    trustAnchors.add(rootCertificate);

    /* We need PKIX parameters to specify the trust anchors
     * and disable the CRL validation
```

```
     */
    validationParameters = new PKIXParameters(trustAnchors);
    certPathValidator = CertPathValidator.getInstance(PKIX);
    validationParameters.setRevocationEnabled(false);

    /* We are ensuring that certificates are chained correctly */
    certPathValidator.validate(certPath, validationParameters);
}
```

## Mengintegrasikan dengan AWS KMS

Instans Anda harus memiliki aplikasi yang dapat membuat permintaan AWS KMS API dengan Dokumen Pengesahan yang diambil dari NitroTPM. Saat Anda membuat permintaan dengan Dokumen Pengesahan, AWS KMS memvalidasi pengukuran dalam Dokumen Pengesahan yang disediakan terhadap pengukuran referensi dalam kebijakan kunci KMS. Permintaan hanya diperbolehkan jika pengukuran dalam Dokumen Pengesahan cocok dengan pengukuran referensi dalam kebijakan kunci KMS.

Saat Anda memanggil operasi Decrypt DeriveSharedSecret, GenerateDataKey, GenerateDataKeyPair,,, atau GenerateRandom API dengan Dokumen Pengesahan, API ini mengenkripsi plaintext dalam respons di bawah kunci publik dari Dokumen Pengesahan, dan mengembalikan ciphertext alih-alih plaintext. Ciphertext ini dapat didekripsi hanya dengan menggunakan kunci pribadi yang cocok yang dihasilkan dalam instance.

Untuk informasi selengkapnya, lihat Penges ahan Kriptografi untuk NitroTPM di Panduan Pengembang. AWS Key Management Service

> **Note**
>
> Jika Anda membuktikan layanan pihak ketiga, Anda harus membangun mekanisme khusus Anda sendiri untuk menerima, mengurai, dan memvalidasi Dokumen Pengesahan. Untuk informasi selengkapnya, lihat Validasi Dokumen Pengesahan NitroTPM.

## Isolasi data dari operator Anda sendiri

Sistem AWS Nitro tidak memiliki akses operator. Tidak ada mekanisme bagi AWS sistem atau orang mana pun untuk masuk ke host Amazon EC2 Nitro, mengakses memori instans EC2, atau mengakses data pelanggan apa pun yang disimpan di penyimpanan instans terenkripsi lokal atau volume Amazon EBS terenkripsi jarak jauh.

Saat memproses data yang sangat sensitif, Anda dapat mempertimbangkan untuk membatasi akses ke data tersebut dengan mencegah operator Anda sendiri mengakses instans EC2.

Anda dapat membuat AMI yang Dapat Diverifikasi kustom yang dikonfigurasi untuk menyediakan lingkungan komputasi terisolasi. Konfigurasi AMI tergantung pada beban kerja dan persyaratan aplikasi Anda. Pertimbangkan praktik terbaik ini saat membangun AMI Anda untuk membuat lingkungan komputasi terisolasi.

- Hapus semua akses interaktif untuk mencegah operator atau pengguna mengakses instance.
- Pastikan hanya perangkat lunak dan kode tepercaya yang disertakan dalam AMI.
- Konfigurasikan firewall jaringan dalam instance untuk memblokir akses.
- Pastikan status read-only dan tidak dapat diubah untuk semua penyimpanan dan sistem file.
- Batasi akses instance ke panggilan API yang diautentikasi, diotorisasi, dan dicatat.

## Memperbarui AMI yang dapat dibuktikan yang tidak memiliki akses interaktif

Setelah Anda meluncurkan instance menggunakan AMI lingkungan komputasi terisolasi, tidak ada cara bagi pengguna atau operator mana pun untuk terhubung ke instans. Ini berarti bahwa tidak ada cara untuk menginstal atau memperbarui perangkat lunak apa pun pada instance setelah peluncuran.

Jika perangkat lunak baru atau pembaruan perangkat lunak diperlukan, Anda harus membuat AMI yang dapat dibuktikan baru yang menyertakan pembaruan perangkat lunak atau perangkat lunak yang diperlukan. Kemudian, gunakan AMI tersebut untuk meluncurkan instance baru, atau untuk melakukan penggantian volume root pada instance asli. Setiap perubahan perangkat lunak yang dilakukan pada AMI akan menghasilkan hash baru yang dihasilkan.

Tindakan berikut akan menghasilkan perubahan pada pengukuran referensi dalam Dokumen Pengesahan NitroTPM:

- Menghentikan dan memulai instance yang diluncurkan dengan AMI yang Dapat Dibuktikan
- Melakukan penggantian volume root dengan AMI yang berbeda

Jika Anda melakukan salah satu tindakan ini, Anda harus memperbarui layanan pengesahan Anda dengan pengukuran referensi baru. Misalnya, Anda harus memperbarui kebijakan kunci KMS ke pengukuran referensi baru jika Anda menggunakan AWS KMS untuk pengesahan.

Sebuah instance mempertahankan materi kunci NitroTPM-nya untuk seluruh siklus hidup instans, dan tetap bertahan melalui operasi stop/starts penggantian volume root.

# Credential Guard untuk instans Windows

Sistem AWS Nitro mendukung Credential Guard untuk instans Windows Amazon Elastic Compute Cloud (Amazon EC2). Credential Guard adalah fitur keamanan berbasis virtualisasi (VBS) Windows yang memungkinkan pembuatan lingkungan yang terisolasi untuk melindungi aset keamanan, seperti kredensial pengguna Windows dan pemberlakuan integritas kode, di luar perlindungan kernel Windows. Saat Anda menjalankan instance EC2 Windows, Credential Guard menggunakan Sistem AWS Nitro untuk melindungi kredenSIAL login Windows agar tidak diekstraksi dari memori sistem operasi.

Daftar Isi


- Prasyarat
- Luncurkan instance yang didukung
- Nonaktifkan integritas memori
- Aktifkan Credential Guard
- Verifikasi bahwa Credential Guard sedang berjalan


## Prasyarat

Instans Windows Anda harus memenuhi prasyarat berikut untuk menggunakan Credential Guard.

Amazon Machine Image (AMI)

AMI harus dikonfigurasikan sebelumnya untuk mengaktifkan Secure Boot NitroTPM dan UEFI. Untuk informasi selengkapnya tentang AMI yang didukung, lihatthe section called “Persyaratan”.

Integritas memori

Integritas memori, juga dikenal sebagai integritas kode yang dilindungi hypervisor (HVCI) atau integritas kode yang diberlakukan hypervisor, tidak didukung. Sebelum Anda mengaktifkan Credential Guard, Anda harus memastikan fitur ini dinonaktifkan. Untuk informasi selengkapnya, lihat Nonaktifkan integritas memori.

Tipe instans

Jenis instans berikut mendukung Credential Guard di semua ukuran kecuali dinyatakan lain:`C5`,,`C5d`,`C5n`,`C6i`,`C6id`,`C6in`,`C7i`,,`C7i-flex`,`M5`,`M5d`,`M5dn`,`M5n`,`M5zn`,,`M6i`,`M6id`,`M6idn`,`M6in`,`M7i`,`M7i-flex`,`R5`,,`R5b`,`R5d`,`R5dn`,`R5n`,`R6i`,`R6id`,,`R6idn`, `R6in` `R7iR7iz`,`T3`.

> **Note**
>
> - Meskipun NitroTPM memiliki beberapa jenis instans yang diperlukan yang sama, jenis instans harus menjadi salah satu jenis instans sebelumnya untuk mendukung Credential Guard.
> - Credential Guard tidak didukung untuk:
>   - Contoh logam telanjang.
>   - Jenis contoh berikut:`C7i.48xlarge`,`M7i.48xlarge`, dan`R7i.48xlarge`.

Untuk informasi selengkapnya tentang jenis instans, lihat Panduan Jenis Instans Amazon EC2.

## Luncurkan instance yang didukung

Anda dapat menggunakan konsol Amazon EC2 atau AWS Command Line Interface (AWS CLI) untuk meluncurkan instans yang dapat mendukung Credential Guard. Anda akan memerlukan ID AMI yang kompatibel untuk meluncurkan instans Anda yang unik untuk setiap Wilayah AWS.

> **Tip**
>
> Anda dapat menggunakan tautan berikut untuk menemukan dan meluncurkan instans dengan AMI yang disediakan Amazon yang kompatibel di konsol Amazon EC2: https://console.aws.amazon.com/ec2/v2/home?#Images:visibility=public-images;v=3;search=:TPM-Windows_Server;ownerAlias=amazon

Console

Untuk meluncurkan sebuah instans

Ikuti langkah-langkah untuk meluncurkan instance, menentukan jenis instance yang didukung dan AMI Windows yang telah dikonfigurasi sebelumnya.

AWS CLI

Untuk meluncurkan sebuah instans

Gunakan perintah run-instances untuk meluncurkan instans menggunakan tipe instans yang didukung dan AMI Windows yang telah dikonfigurasi sebelumnya.

```
aws ec2 run-instances \
    --image-id resolve:ssm:/aws/service/ami-windows-latest/TPM-Windows_Server-2022-English-Full-Base \
    --instance-type c6i.large \
    --region us-east-1 \
    --subnet-id subnet-0abcdef1234567890
    --key-name key-name
```

PowerShell

Untuk meluncurkan sebuah instans

Gunakan perintah New-EC2Instance untuk meluncurkan instans menggunakan tipe instans yang didukung dan AMI Windows yang telah dikonfigurasi sebelumnya.

```
New-EC2Instance `
    -ImageId resolve:ssm:/aws/service/ami-windows-latest/TPM-Windows_Server-2022-English-Full-Base `
    -InstanceType c6i.large `
    -Region us-east-1 `
    -SubnetId subnet-0abcdef1234567890 `
    -KeyName key-name
```

## Nonaktifkan integritas memori

Anda dapat menggunakan Editor Kebijakan Grup Lokal untuk menonaktifkan integritas memori dalam skenario yang didukung. Panduan berikut dapat diterapkan untuk setiap pengaturan konfigurasi di bawah Perlindungan Integritas Kode Berbasis  Virtualisasi:

- Diaktifkan tanpa kunci – Ubah pengaturan ke Dinonaktifkan untuk menonaktifkan integritas memori.
- Diaktifkan dengan kunci UEFI – Integritas memori telah diaktifkan dengan kunci UEFI. Integritas memori tidak dapat dinonaktifkan setelah diaktifkan dengan kunci UEFI. Sebaiknya buat instans baru dengan integritas memori dinonaktifkan dan menghentikan instans yang tidak didukung jika tidak digunakan.

Untuk menonaktifkan integritas memori dengan Editor Kebijakan Grup Lokal

1. Terhubung ke instans Anda sebagai akun pengguna dengan hak akses administrator menggunakan Protokol Desktop Jarak Jauh (RDP). Untuk informasi selengkapnya, lihat the section called “Hubungkan menggunakan klien RDP”.
2. Buka menu Mulai dan cari **cmd** untuk memulai prompt perintah.
3. Jalankan perintah berikut untuk membuka Editor Kebijakan Grup Lokal: `gpedit.msc`
4. Di Editor Kebijakan Grup Lokal, pilih Konfigurasi Komputer, Templat Administratif, Sistem, Penjaga Perangkat.
5. Pilih Aktifkan Keamanan Berbasis Virtualisasi, lalu pilih Edit pengaturan kebijakan.
6. Buka drop-down pengaturan untuk Perlindungan Integritas Kode Berbasis Virtualisasi, pilih Nonaktifkan, lalu pilih Terapkan.
7. Boot ulang instans untuk menerapkan perubahan.

## Aktifkan Credential Guard

Setelah meluncurkan instans Windows dengan tipe instans yang didukung dan AMI yang kompatibel dan mengonfirmasi bahwa integritas memori dinonaktifkan, Anda dapat mengaktifkan Credential Guard.

> **Important**
>
> Hak akses administrator diperlukan untuk melakukan langkah-langkah berikut guna mengaktifkan Credential Guard.

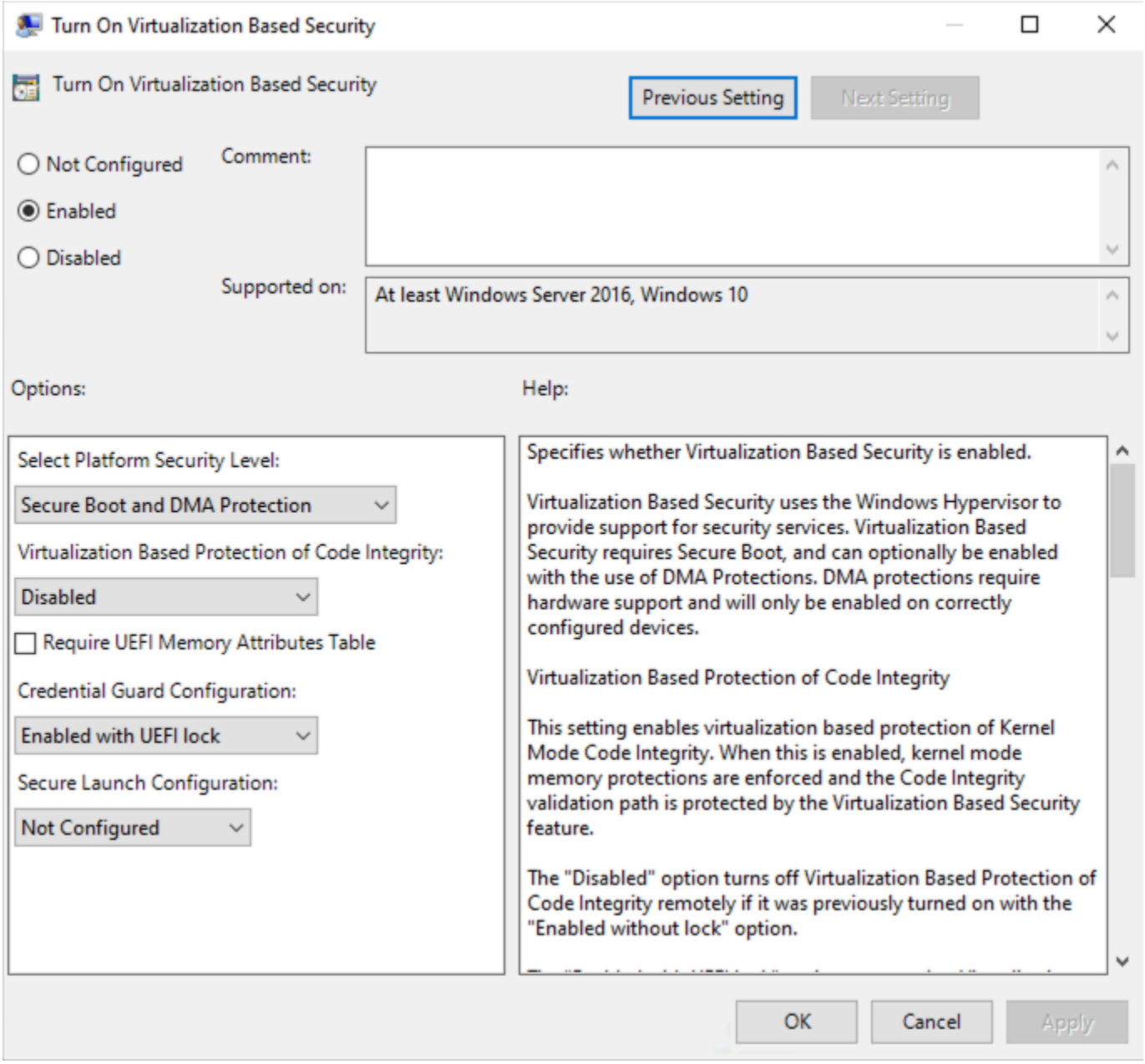
Mengaktifkan Credential Guard

1. Terhubung ke instans Anda sebagai akun pengguna dengan hak akses administrator menggunakan Protokol Desktop Jarak Jauh (RDP). Untuk informasi selengkapnya, lihat the section called “Hubungkan menggunakan klien RDP”.
2. Buka menu Mulai dan cari **cmd** untuk memulai prompt perintah.
3. Jalankan perintah berikut untuk membuka Editor Kebijakan Grup Lokal: `gpedit.msc`
4. Di Editor Kebijakan Grup Lokal, pilih Konfigurasi Komputer, Templat Administratif, Sistem, Penjaga Perangkat.

5. Pilih Aktifkan Keamanan Berbasis Virtualisasi, lalu pilih Edit pengaturan kebijakan.
6. Pilih Diaktifkan dalam menu Aktifkan Keamanan Berbasis Virtualisasi.
7. Untuk Pilih Tingkat Keamanan Platform, pilih Secure Boot dan Perlindungan DMA.
8. Untuk Konfigur asi Penjaga Kredensi, pilih Di aktifkan dengan kunci UEFI.

> **Note**
>
> Pengaturan kebijakan yang tersisa tidak diperlukan untuk mengaktifkan Credential Guard dan dapat dibiarkan sebagai Tidak Dikonfigurasikan.

Gambar berikut menampilkan pengaturan VBS yang dikonfigurasikan seperti yang dijelaskan sebelumnya:




9. Boot ulang instans untuk menerapkan pengaturan.

## Verifikasi bahwa Credential Guard sedang berjalan

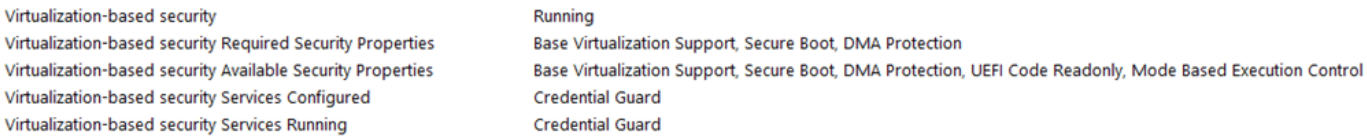
Anda dapat menggunakan alat Informasi Sistem Microsoft (`Msinfo32.exe`) untuk mengonfirmasi bahwa Credential Guard sedang berjalan.

> **Important**
>
> Anda harus melakukan boot ulang instans terlebih dahulu untuk menyelesaikan penerapan pengaturan kebijakan yang diperlukan untuk mengaktifkan Credential Guard.

Untuk memverifikasi bahwa Credential Guard sedang berjalan

1. Hubungkan ke instans Anda menggunakan Protokol Desktop Jarak Jauh (RDP). Untuk informasi selengkapnya, lihat the section called “Hubungkan menggunakan klien RDP”.
2. Dalam sesi RDP ke instans Anda, buka menu Mulai dan cari **`cmd`** untuk memulai prompt perintah.
3. Buka Informasi Sistem dengan menjalankan perintah berikut: `msinfo32.exe`
4. Alat Informasi Sistem Microsoft mencantumkan detail untuk konfigurasi VBS. Di sebelah Layanan Virtualization-based keamanan, konfirmasikan bahwa Credential Guard muncul sebagai Run ning.

   Gambar berikut menampilkan VBS berjalan seperti yang dijelaskan sebelumnya:




# Mengakses Amazon EC2 menggunakan titik akhir VPC antarmuka

Anda dapat meningkatkan postur keamanan VPC Anda dengan membuat koneksi pribadi antara sumber daya di VPC Anda dan Amazon EC2 API. Anda dapat mengakses Amazon EC2 API seolah-olah berada di VPC Anda, tanpa menggunakan gateway internet, perangkat NAT, koneksi VPN, atau Direct Connect koneksi. Instans EC2 di VPC Anda tidak memerlukan alamat IP publik untuk mengakses Amazon EC2 API.

Untuk informasi selengkapnya, lihat Ak AWS PrivateLink ses Layanan AWS melalui dalam AWS PrivateLink Panduan.

Daftar Isi


- Membuat titik akhir VPC antarmuka
- Membuat kebijakan titik akhir

## Membuat titik akhir VPC antarmuka

Buatlah titik akhir antarmuka untuk Amazon EC2 menggunakan nama layanan berikut:

- com.amazonaws. ***region***.ec2 — Membuat titik akhir untuk tindakan Amazon EC2 API.

Untuk informasi selengkapnya, lihat Meng akses Layanan AWS menggunakan titik akhir VPC antarmuka di AWS PrivateLink Panduan.

## Membuat kebijakan titik akhir

kebijakan titik akhir adalah sumber daya IAM yang dapat Anda lampirkan ke titik akhir antarmuka Anda. Kebijakan titik akhir default mengizinkan akses penuh ke Amazon EC2 API melalui titik akhir antarmuka. Untuk mengontrol akses yang diizinkan ke API Amazon EC2 dari VPC Anda, lampirkan kebijakan titik akhir kustom ke titik akhir antarmuka.

kebijakan titik akhir mencantumkan informasi berikut:

- prinsipal utama yang dapat melakukan tindakan.
- Tindakan-tindakan yang dapat dilakukan.
- Sumber daya yang padanya tindakan dapat dilakukan.

> **Important**
>
> Saat kebijakan non-default diterapkan ke titik akhir VPC antarmuka untuk Amazon EC2, permintaan API tertentu yang gagal, seperti permintaan yang gagal`RequestLimitExceeded`, mungkin tidak masuk ke AWS CloudTrail atau Amazon. CloudWatch

Untuk informasi selengkapnya, lihat Mengontrol akses ke layanan menggunakan kebijakan titik akhir di Panduan AWS PrivateLink .

Contoh berikut menunjukkan kebijakan titik akhir VPC yang menolak izin untuk membuat volume yang tidak terenkripsi atau untuk meluncurkan instans yang memiliki volume yang tidak terenkripsi. Contoh kebijakan tersebut juga memberikan izin untuk melakukan semua tindakan Amazon EC2 lainnya.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
    {
        "Action": "ec2:*",
        "Effect": "Allow",
        "Resource": "*",
        "Principal": "*"
    },
    {
        "Action": [
            "ec2:CreateVolume"
        ],
        "Effect": "Deny",
        "Resource": "*",
        "Principal": "*",
        "Condition": {
            "Bool": {
                "ec2:Encrypted": "false"
            }
        }
    },
    {
        "Action": [
            "ec2:RunInstances"
        ],
        "Effect": "Deny",
        "Resource": "*",
        "Principal": "*",
        "Condition": {
            "Bool": {
                "ec2:Encrypted": "false"
            }
        }
    }]
}
```

# Opsi penyimpanan untuk instans Amazon EC2 Anda

Amazon EC2 memberi Anda opsi penyimpanan data yang fleksibel, hemat biaya, dan mudah digunakan untuk instans Anda. Setiap pilihan memiliki kombinasi performa dan daya tahan yang unik. Opsi penyimpanan ini dapat digunakan secara independen atau bersamaan untuk menyesuaikan kebutuhan Anda.

Blok penyimpanan

- Amazon EBS — Amazon EBS menyediakan volume penyimpanan tingkat blok yang tahan lama yang dapat Anda lampirkan dan lepaskan dari instans Anda. Anda dapat memasang beberapa volume EBS ke suatu instans. Volume EBS berlanjut secara terpisah dari masa aktif instans terkait. Anda dapat mengenkripsi volume EBS Anda. Untuk menyimpan salinan cadangan data, Anda dapat membuat snapshot dari volume EBS Anda. Snapshot disimpan di Amazon S3. Anda dapat membuat volume EBS dari snapshot.
- Penyimpanan blok sementara instans untuk instans EC2— Toko instans menyediakan penyimpanan tingkat blok sementara untuk instance. Jumlah, ukuran, dan tipe volume penyimpanan instans ditentukan oleh tipe instans dan ukuran instans. Data pada suatu volume penyimpanan instans hanya akan berlanjut selama masa pakai instans yang terkait; jika Anda berhenti, melakukan hibernasi, atau mengakhiri suatu instans, data apa pun yang berupa volume penyimpanan instan akan hilang.

Penyimpanan objek

- Amazon S3- Amazon S3 menyediakan akses ke infrastruktur penyimpanan data yang andal dan murah. Didesain untuk memudahkan komputasi skala web dengan memungkinkan Anda menyimpan dan mengambil data dalam jumlah berapa pun, kapan pun, dari Amazon EC2 atau di mana pun di web. Misalnya, Anda dapat menggunakan Amazon S3 untuk menyimpan salinan cadangan data dan aplikasi Anda. Amazon EC2 menggunakan Amazon S3 untuk menyimpan snapshot EBS dan AMI Amazon. S3-backed

Penyimpanan file

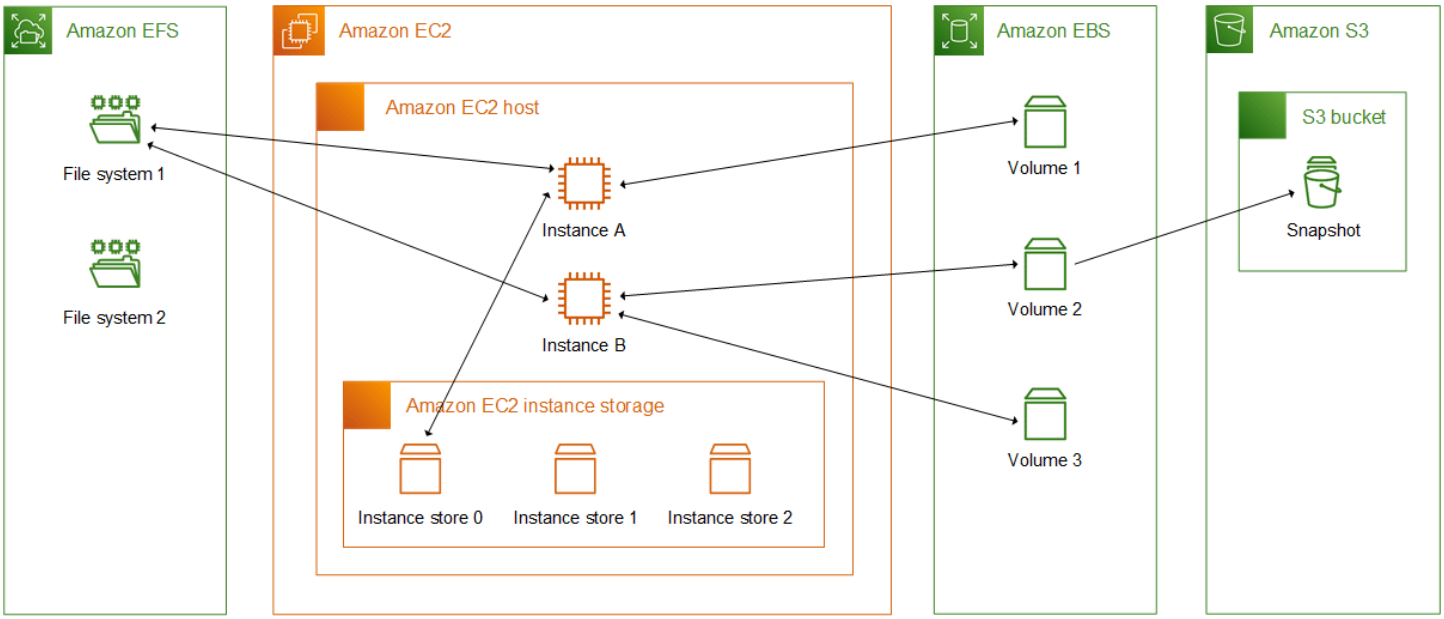
- Amazon EFS(Hanya instans Linux) - Amazon EFS menyediakan penyimpanan file yang dapat diskalakan untuk digunakan dengan Amazon EC2. Anda dapat membuat sistem file EFS dan mengonfigurasikan instans Anda untuk memasang sistem file. Anda dapat menggunakan sistem file EFS sebagai sumber data umum untuk beban kerja dan aplikasi yang berjalan pada beberapa instans.
- Amazon FSx— Dengan Amazon FSx, Anda dapat meluncurkan, menjalankan, dan menskalakan sistem file berkinerja tinggi yang kaya fitur di cloud. Amazon FSx adalah layanan

terkelola penuh yang mendukung berbagai beban kerja. Anda dapat memilih antara sistem file yang banyak digunakan ini: Lustre, NetApp ONTAP, OpenZFS, dan Windows File Server.

Caching file

- Menggunakan Cache File Amazon dengan EC2 instans Amazon- Amazon File Cache menyediakan cache sementara berkinerja tinggi AWS untuk memproses data file. Cache menyediakan akses data baca dan tulis untuk menghitung beban kerja di Amazon EC2 dengan latensi sub-milidetik, hingga ratusan throughput, dan hingga GB/s jutaan IOPS.

Gambar berikut ini menunjukkan hubungan antara opsi penyimpanan ini dan instans Anda.




# AWS Harga penyimpanan

Buka AWS Harga, gulir ke Harga untuk AWS produk dan pilih Penyimpanan. Pilih produk penyimpanan untuk membuka halaman harga.

# Penyimpanan blok persisten Amazon EBS untuk instans Amazon EC2

Amazon Elastic Block Store (Amazon EBS) menyediakan sumber daya penyimpanan blok yang dapat diskalakan dan berkinerja tinggi yang dapat digunakan dengan instans Amazon EC2. Dengan Amazon EBS, Anda dapat membuat dan mengelola sumber daya penyimpanan blok berikut:

- Volume Amazon EBS — Ini adalah volume penyimpanan yang Anda lampirkan ke instans Amazon EC2. Setelah Anda melampirkan volume ke instance, Anda dapat menggunakannya dengan cara yang sama seperti Anda menggunakan penyimpanan blok. Instance dapat berinteraksi dengan volume seperti halnya dengan drive lokal.
- Snapshot Amazon EBS — Ini adalah cadangan point-in-time volume Amazon EBS yang bertahan secara independen dari volume itu sendiri. Anda dapat membuat snapshot untuk membuat cadangan data pada volume Amazon EBS Anda. Anda kemudian dapat memulihkan volume baru dari snapshot tersebut kapan saja.

Anda dapat membuat dan melampirkan volume EBS ke instans selama peluncuran, dan Anda dapat membuat dan melampirkan volume EBS ke instans kapan saja setelah peluncuran. Anda juga dapat meningkatkan ukuran atau kinerja volume EBS tanpa melepaskan volume atau memulai ulang instans Anda.

Anda dapat membuat snapshot EBS dari volume EBS kapan saja setelah pembuatan. Anda dapat menggunakan snapshot EBS untuk membuat cadangan data yang tersimpan di volume Anda. Anda kemudian dapat menggunakan snapshot tersebut untuk memulihkan volume secara instan, atau untuk memigrasikan data di seluruh Akun AWS, Wil AWS ayah, atau Zona Ketersediaan. Anda dapat menggunakan Amazon Data Lifecycle Manager atau AWS Backup untuk mengotomatiskan pembuatan, penyimpanan, dan penghapusan snapshot EBS Anda.

Volume EBS yang dikelola dikelola oleh penyedia layanan, seperti Mode Otomatis Amazon EKS. Anda tidak dapat secara langsung mengubah pengaturan volume EBS terkelola. Volume EBS yang dikelola diidentifikasi dengan nilai sebenarnya di bidang Terkel ola. Untuk informasi selengkapnya, lihat Instans terkelola Amazon EC2.

Untuk informasi selengkapnya tentang bekerja dengan volume dan snapshot, lihat Panduan Pengguna Amazon EBS.

## Batas volume Amazon EBS untuk instans Amazon EC2

Jumlah maksimum volume Amazon EBS yang dapat dilampirkan ke instans bergantung pada tipe instans dan ukuran instans. Jika Anda melebihi batas lampiran volume untuk sebuah instance, permintaan lampiran gagal dengan `AttachmentLimitExceeded` kesalahan.

Saat mempertimbangkan berapa banyak volume yang akan dilampirkan ke instans Anda, Anda harus mempertimbangkan apakah Anda memerlukan peningkatan I/O bandwidth atau peningkatan kapasitas penyimpanan.

Bandwidth versus kapasitas

Untuk kasus penggunaan bandwidth yang konsisten dan dapat diprediksi, gunakan EBS-optimized instans Amazon dengan volume SSD Tujuan Umum atau volume SSD IOPS yang disediakan. Untuk performa maksimum, cocokkan IOPS yang telah Anda sediakan untuk volume Anda dengan bandwidth yang tersedia untuk tipe instans Anda.

Untuk konfigurasi RAID, Anda mungkin menemukan bahwa array yang lebih besar dari 8 volume telah mengurangi pengembalian kinerja karena peningkatan I/O overhead. Uji performa aplikasi individual Anda dansesuai kebutuhan.

Daftar Isi


- Batas volume untuk instans yang dibangun di atas Sistem Nitro
  - Batas volume EBS khusus
  - Batas volume EBS bersama
- Batas volume untuk Xen-based instance
  - Instans Linux
  - Instans Windows


## Batas volume untuk instans yang dibangun di atas Sistem Nitro

Batas volume untuk instans yang dibangun di Nitro System bergantung pada jenis instans. Beberapa jenis instans Nitro memiliki batas volume EBS  khusus, sementara sebagian besar memiliki batas volume  bersama.  Untuk melihat batas lampiran volume untuk setiap jenis instans, lihat berikut ini:

- Spesifikasi Amazon EBS — Tujuan umum
- Spesifikasi Amazon EBS — Komputasi dioptimalkan
- Spesifikasi Amazon EBS - Memori dioptimalkan
- Spesifikasi Amazon EBS — Penyimpanan dioptimalkan
- Spesifikasi Amazon EBS — Komputasi yang dipercepat
- Spesifikasi Amazon EBS — komputasi High-performance
- Spesifikasi Amazon EBS — Generasi sebelumnya

### Batas volume EBS khusus

Jenis instans Nitro berikut memiliki batas volume EBS khusus yang bervariasi tergantung pada ukuran instans. Batas tidak dibagikan dengan lampiran perangkat lain. Dengan kata lain, Anda dapat melampirkan sejumlah volume EBS hingga batas lampiran volume, terlepas dari jumlah perangkat yang terpasang, seperti volume penyimpanan instans NVMe dan antarmuka jaringan.

- Tujuan umum: M7a | M7i | | M8a | M8azN | M8g M7i-flex | M8gD | M8gD | M8gn | M8i | M8id | | M8in | M8idn | M8iDN | M8iB | M8iDB | M9g | M9gd M8i-flex
- Komputasi dioptimalkan: C7a | C7i | | C8a | C8g | C8gD C7i-flex | C8gD | C8gN | C8i | C8id | | C8in | C8iN | C8iB | C9g | C9gd C8i-flex
- Memori dioptimalkan: R7a | R7i | R7iz | R8a | R8g | R8gD | R8gD | R8gn | R8i | R8id | | R8in | R8idn | R8ib | R8iDB | R9g | R9gd | | | | | | X8g R8i-flex | X8aedz | X8i U7i-6tb U7i-8tb U7i-12tb U7in-16tb U7in-24tb U7in-32tb U7inh-32tb
- Penyimpanan dioptimalkan: i7i | i7IE | i8G | i8GE
- Komputasi yang dipercepat: F2 | G6 | G6e | G6f | Gr6 | Gr6f | G7 | G7e | P4d | P4de | P5 | P5e | P5en | | | | Trn2 | Trn2u P6-B200 P6-B300 P6e-GB200
- Komputasi kinerja tinggi: HPC7a | HPC8a

### Batas volume EBS bersama

Semua jenis instans Nitro lainnya (tidak tercantum di[Batas volume EBS khusus](#)) memiliki batas lampiran volume yang dibagi antara volume Amazon EBS, antarmuka jaringan, dan volume penyimpanan instans NVMe. Anda dapat melampirkan sejumlah volume Amazon EBS hingga batas tersebut, dikurangi jumlah antarmuka jaringan yang dilampirkan dan volume penyimpanan instans NVMe. Ingatlah bahwa setiap instans harus memiliki setidaknya satu antarmuka jaringan, dan volume penyimpanan instans NVMe dilampirkan secara otomatis saat peluncuran.

Sebagian besar instans Nitro mendukung maksimal 28 lampiran. Contoh berikut menunjukkan cara menghitung berapa banyak volume EBS yang dapat Anda lampirkan.

Contoh

- Dengan `m5.xlarge` instance dengan hanya antarmuka jaringan utama, Anda dapat melampirkan 27 volume EBS.

  28 volume - 1 antarmuka jaringan = 27

- Dengan `m5.xlarge` instance dengan dua antarmuka jaringan tambahan, Anda dapat melampirkan 25 volume EBS.

  28 volume - 3 antarmuka jaringan = 25
- Dengan `m5d.xlarge` instance dengan dua antarmuka jaringan tambahan, Anda dapat melampirkan 24 volume EBS.

  28 volume - 3 antarmuka jaringan - 1 volume penyimpanan instans NVMe = 24

## Batas volume untuk Xen-based instance

Batas volume untuk Xen-based instance, seperti T2, tergantung pada sistem operasi.

Untuk informasi selengkapnya, lihat Xen-based instance.

### Instans Linux

Melampirkan lebih dari 40 volume ke instance Xen-based Linux dapat menyebabkan kegagalan boot. Angka ini mencakup volume root, ditambah volume penyimpanan instans terlampir dan volume Amazon EBS.

Jika Anda mengalami masalah booting pada suatu instans dengan jumlah volume yang banyak, hentikan instans, lepaskan semua volume yang tidak penting dalam proses booting, mulai instans, lalu pasang kembali volume setelah instans berjalan.

> **Important**
>
> Melampirkan lebih dari 40 volume ke instance Xen-based Linux didukung atas dasar upaya terbaik saja dan tidak dijamin.

### Instans Windows

Tabel berikut menunjukkan batas volume untuk inst Xen-based ans Windows berdasarkan driver yang digunakan. Angka-angka ini termasuk volume root, ditambah volume penyimpanan instans dan volume Amazon EBS yang terlampir.

| Driver | Batas Volume |
| --- | --- |
| AWS PV | 26 |

| Driver | Batas Volume |
| --- | --- |
| Citrix PV | 26 |
| Red Hat PV | 17 |

Kami menyarankan Anda untuk tidak melampirkan lebih dari 26 volume ke instance Xen-based Windows dengan driver AWS PV atau Citrix PV, karena kemungkinan akan menyebabkan masalah kinerja. Untuk menentukan driver PV mana yang digunakan oleh instans Anda, atau untuk meningkatkan instans Windows Anda dari Red Hat ke driver Citrix PV, lihat the section called “Mutakhirkan driver PV”.

> **Important**
>
> Melampirkan lebih dari jumlah volume berikut ke instance Xen-based Windows didukung berdasarkan upaya terbaik saja dan tidak dijamin.

Untuk informasi selengkapnya tentang cara nama perangkat terkait volume, lihat Cara volume dilampirkan dan dipetakan untuk instans Windows Amazon EC2.

## Kartu EBS

Sebagian besar jenis instans mendukung satu kartu EBS. Jenis instans yang mendukung beberapa kartu EBS memberikan EBS-optimized kinerja yang lebih tinggi, baik dalam throughput EBS maupun IOPS. Performa maksimum instans Amazon EC2 Anda tersebar merata di setiap kartu EBS. Misalnya, pada instans EC2 yang mendukung `1,000,000` total IOPS dengan 2 kartu EBS, setiap kartu EBS dapat mendukung hingga IOPS. `500,000` Untuk informasi tentang kinerja Amazon EBS yang didukung dari instans Amazon EC2 Anda, lihat. Jenis EBS-optimized instans Amazon

Saat Anda melampirkan volume Amazon EBS ke instans yang mendukung beberapa kartu EBS, Anda dapat memilih kartu EBS untuk volume dengan menentukan indeks kartu EBS. Volume root harus dilampirkan ke indeks kartu EBS 0.

## Jenis instans dengan beberapa kartu EBS

Jenis instance berikut mendukung beberapa kartu EBS. Untuk informasi tentang jumlah volume Amazon EBS yang didukung oleh jenis instans, lihatBatas volume Amazon EBS untuk instans Amazon EC2.

| Tipe Instans | Jumlah kartu EBS |
|---|---|
| Tujuan Umum | |
| m8gb.48xbesar | 2 |
| m8gb.logam-48xl | 2 |
| m8gn.48xbesar | 2 |
| m8gn.logam-48xl | 2 |
| m8in.96xbesar | 2 |
| m8in.logam-96xl | 2 |
| m8idn.96xbesar | 2 |
| m8idn.logam-96xl | 2 |
| m8ib.96xbesar | 2 |
| m8ib.logam-96xl | 2 |
| m8idb.96xbesar | 2 |
| m8idb.logam-96xl | 2 |
| Komputasi Dioptimalkan | |
| c8gb.48xbesar | 2 |
| c8gb.logam-48xl | 2 |
| c8gn.48xbesar | 2 |

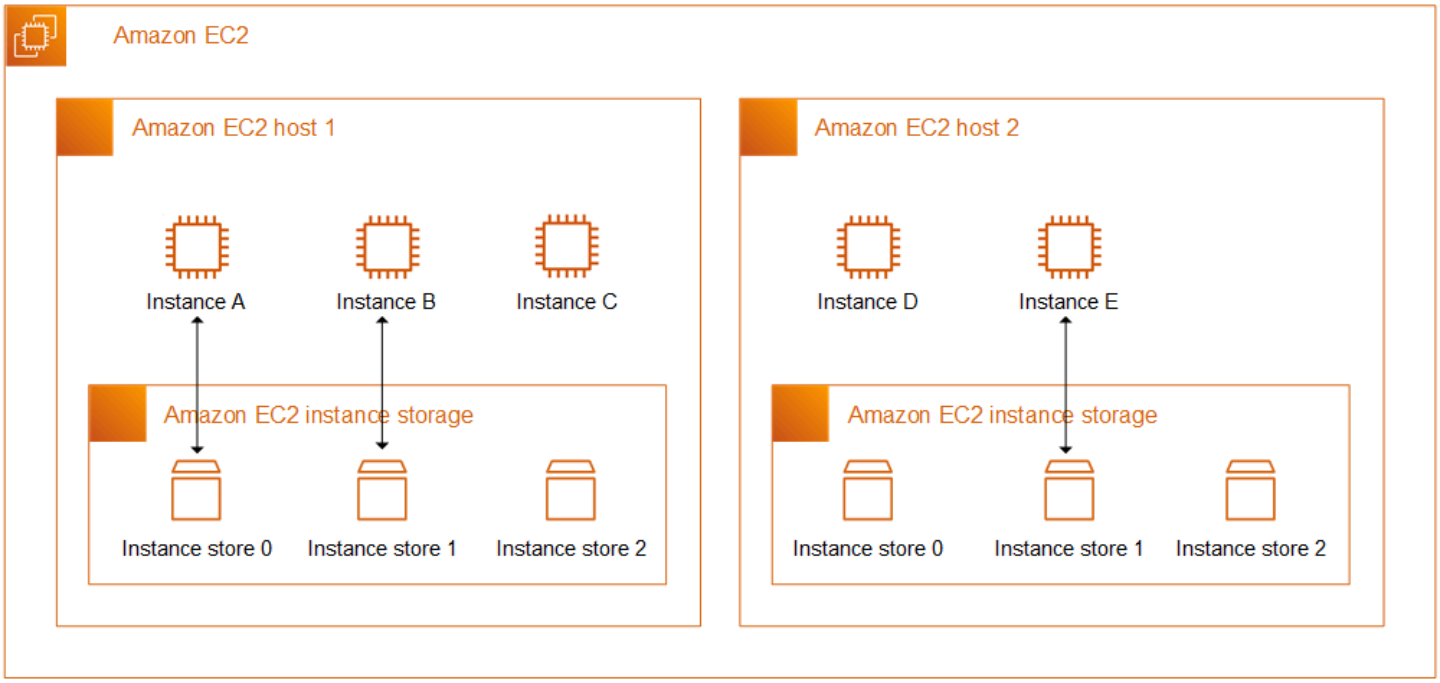
| Tipe Instans | Jumlah kartu EBS |
| --- | --- |
| c8gn.logam-48xl | 2 |
| c8in.96xbesar | 2 |
| c8in.logam-96xl | 2 |
| c8ib.96xbesar | 2 |
| c8ib.logam-96xl | 2 |
| Memori Dioptimalkan | |
| r8gb.48xbesar | 2 |
| r8gb.logam-48xl | 2 |
| r8gn.48xbesar | 2 |
| r8gn.logam-48xl | 2 |
| r8in.96xbesar | 2 |
| r8in.logam-96xl | 2 |
| r8idn.96xbesar | 2 |
| r8idn.logam-96xl | 2 |
| r8ib.96xbesar | 2 |
| r8ib.logam-96xl | 2 |
| r8idb.96xbesar | 2 |
| r8idb.logam-96xl | 2 |
| Komputasi yang Dipercepat | |
| g7.48xbesar | 2 |

# Penyimpanan blok sementara instans untuk instans EC2

Penyimpanan instans menyediakan penyimpanan tingkat blok sementara untuk instans EC2 Anda. Penyimpanan ini disediakan oleh disk yang secara fisik terpasang ke komputer host. Penyimpanan instans ideal untuk penyimpanan sementara informasi yang sering berubah, seperti buffer, cache, data awal, dan konten sementara lainnya. Ini juga dapat digunakan untuk menyimpan data sementara yang Anda replikasi di seluruh armada instans, seperti kumpulan server web yang seimbang dengan beban.

Penyimpanan instans terdiri dari satu atau lebih volume penyimpanan instans yang terekspos sebagai perangkat blok. Ukuran penyimpanan instans serta jumlah perangkat yang tersedia bervariasi berdasarkan tipe instans dan ukuran instans. Misalnya, tidak setiap jenis instans menyediakan volume penyimpanan instans. Untuk informasi selengkapnya, lihat Batas volume penyimpanan instans untuk instans EC2.

Perangkat virtual misalnya volume penyimpanan diberi nama perangkat virtual dalam urutan dari `ephemeral0` ke`ephemeral23`. Misalnya, dengan tipe instance yang mendukung satu volume penyimpanan instance, nama perangkat virtual dari satu volume adalah`ephemeral0`. Dengan tipe instance yang mendukung empat volume penyimpanan instans, nama perangkat virtual dari empat volume adalah sebagai berikut:`ephemeral0`,`ephemeral1`, `ephemeral2` dan`ephemeral3`.




Harga penyimpanan instans

Tidak ada biaya tambahan untuk menggunakan volume penyimpanan instans yang disediakan untuk instans Anda. Volume penyimpanan instans disertakan sebagai bagian dari biaya penggunaan instance.

Daftar Isi


- Persistensi data untuk volume penyimpanan instans Amazon EC2
- Batas volume penyimpanan instans untuk instans EC2
- Volume penyimpanan instans SSD untuk instans EC2
- Menambahkan volume penyimpanan instans ke instans EC2
- Aktifkan volume swap penyimpanan instans untuk instans M1 dan C1 EC2
- Inisialisasi volume penyimpanan instans pada instans EC2
- Statistik kinerja terperinci untuk volume penyimpanan instans Amazon EC2


## Persistensi data untuk volume penyimpanan instans Amazon EC2

Volume penyimpanan instans dilampirkan hanya pada peluncuran instans. Anda tidak dapat memasang volume penyimpanan instans setelah peluncuran. Anda tidak dapat melepaskan volume penyimpanan instans dari satu instans dan memasangnya ke instans yang berbeda.

Volume penyimpanan instans hanya ada selama masa pakai instans yang dilampirkan. Anda tidak dapat mengonfigurasi volume penyimpanan instans agar bertahan melebihi masa pakai instans terkait.

Data pada volume penyimpanan instans tetap ada meskipun instans di-boot ulang. Namun, data tidak bertahan jika instans dihentikan, dihibernasi, atau dihentikan. Saat instans dihentikan, dihibernasi, atau diakhiri, setiap blok volume penyimpanan instans dihapus secara kriptografis.

Oleh karena itu, jangan bergantung pada volume penyimpanan instans untuk data jangka panjang yang berharga. Jika Anda perlu mempertahankan data yang disimpan pada volume penyimpanan instans di luar masa pakai instans, Anda perlu menyalin data tersebut secara manual ke penyimpanan yang lebih persisten, seperti volume Amazon EBS, bucket Amazon S3, atau sistem file Amazon EFS.

Ada beberapa peristiwa yang dapat mengakibatkan data Anda tidak bertahan sepanjang masa instans. Tabel berikut menunjukkan apakah data pada volume penyimpanan instans dipertahankan selama peristiwa tertentu, baik untuk instans virtualisasi maupun bare metal.

| Peristiwa | Apa yang terjadi pada data Anda? |
| --- | --- |
| User-initiated peristiwa siklus hidup instance | |
| Instance di-boot ulang | Data tetap ada |
| Instance dihentikan | Data tidak bertahan |
| Instance berhibernasi | Data tidak bertahan |
| Instance dihentikan | Data tidak bertahan |
| Jenis instance diubah | Data tidak bertahan * |
| EBS-backed AMI dibuat dari instance | Data tidak bertahan dalam AMI yang dibuat ** |
| Amazon S3-backed AMI dibuat dari instance (instance Linux) | Data tetap ada dalam bundel AMI yang diunggah ke Amazon S3 *** |
| User-initiated Peristiwa OS | |
| Shutdown dimulai | Data tidak bertahan † |
| Restart dimulai | Data tetap ada |
| AWS acara yang dijadwalkan | |
| Instance berhenti | Data tidak bertahan |
| Reboot instans | Data tetap ada |
| Sistem reboot | Data tetap ada |
| Pensiun instans | Data tidak bertahan |
| Peristiwa yang tidak direncanakan | |
| Pemulihan otomatis yang disederhanakan | Data tidak bertahan |
| CloudWatch pemulihan berbasis tindakan | Data tidak bertahan |
| Disk yang mendasarinya gagal | Data pada disk yang gagal tidak bertahan |

| Peristiwa | Apa yang terjadi pada data Anda? |
| --- | --- |
| Kegagalan listrik | Data tetap ada saat reboot |

* Jika tipe instans baru mendukung penyimpanan instans, instans mendapatkan jumlah volume penyimpanan instans yang didukung oleh tipe instans baru, tetapi data tidak ditransfer ke instans baru. Jika tipe instans baru tidak mendukung penyimpanan instans, instans tidak mendapatkan volume penyimpanan instans.

** Data tidak disertakan dalam EBS-backed AMI, dan tidak disertakan pada volume penyimpanan instans yang dilampirkan ke instans yang diluncurkan dari AMI tersebut.

*** Data disertakan dalam bundel AMI yang diunggah ke Amazon S3. Saat Anda meluncurkan instans dari AMI tersebut, instans mendapatkan volume penyimpanan instans yang dibundel dalam AMI dengan data yang dikandungnya pada saat AMI dibuat.

† Perlindungan penghentian dan perlindungan penghentian tidak melindungi instans terhadap penghentian atau penghentian instans sebagai akibat dari penghentian yang dimulai melalui sistem operasi pada instans. Data yang disimpan pada volume penyimpanan instans tidak bertahan dalam peristiwa penghentian dan penghentian instans.

## Batas volume penyimpanan instans untuk instans EC2

Jumlah, ukuran, dan jenis volume penyimpanan instans ditentukan oleh jenis instans. Beberapa jenis instans, seperti C8i, M8i, dan R8i, tidak mendukung volume penyimpanan instans, sementara jenis instans lainnya, seperti C8id, M8id, dan R8id, mendukung volume penyimpanan instans. Anda tidak dapat melampirkan lebih banyak volume penyimpanan instans ke instans daripada yang didukung oleh tipe instans-nya. Untuk tipe instans yang mendukung volume penyimpanan instans, jumlah dan ukuran volume penyimpanan instans bervariasi menurut ukuran instans. Misalnya, `r8id.large` mendukung volume penyimpanan instans 1 x 118 GB, sementara `r8id.32xlarge` mendukung volume penyimpanan instans 2 x 3800 GB.

Untuk tipe instans dengan volume penyimpanan instans NVMe, semua volume penyimpanan instans yang didukung secara otomatis dilampirkan ke instans saat peluncuran. Misalnya tipe dengan volume penyimpanan instans non-NVMe, seperti C1, C3, M1, M2, M3, R3, D2, H1, I2, X1, dan X1e, Anda harus secara manual menentukan pemetaan perangkat blok untuk volume penyimpanan instans yang ingin Anda lampirkan saat peluncuran. Kemudian, setelah instans diluncurkan, Anda harus memformat dan memasang volume penyimpanan instans terlampir sebelum Anda dapat

menggunakannya. Anda tidak dapat melampirkan volume penyimpanan instans setelah Anda meluncurkan instans tersebut.

Beberapa jenis instance menggunakan NVMe atau SATA-based solid state drive (SSD), sementara yang lain menggunakan SATA-based hard disk drive (HDD). SSD memberikan I/O kinerja acak yang tinggi dengan latensi yang sangat rendah, tetapi Anda tidak memerlukan data untuk bertahan saat instans berakhir atau Anda dapat memanfaatkan arsitektur toleran kesalahan. Untuk informasi selengkapnya, lihat Volume penyimpanan instans SSD untuk instans EC2.

Data volume penyimpanan instans NVMe dan beberapa volume penyimpanan HDD dienkripsi saat istirahat. Untuk informasi selengkapnya, lihat Perlindungan data dalam Amazon EC2.

## Volume penyimpanan instans yang tersedia

Panduan Jenis Instans Amazon EC2 menyediakan jumlah, ukuran, jenis, dan pengoptimalan kinerja volume penyimpanan instans yang tersedia pada setiap jenis instans yang didukung. Untuk informasi selengkapnya, lihat berikut ini:

- Spesifikasi penyimpanan instans — Tujuan umum
- Spesifikasi penyimpanan instans - Komputasi dioptimalkan
- Spesifikasi penyimpanan instans - Memori dioptimalkan
- Spesifikasi penyimpanan instans - Penyimpanan dioptimalkan
- Spesifikasi penyimpanan instans — Komputasi yang dipercepat
- Spesifikasi penyimpanan instans — High-performance komputasi
- Spesifikasi penyimpanan instans - Generasi sebelumnya

Console

Untuk mengambil informasi volume penyimpanan instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Tipe Instans.
3. Tambahkan filter Local instance storage = true. Kol om Storage menunjukkan ukuran total penyimpanan instans untuk jenis instans.
4. (Opsional) Pilih ikon Preferensi dan kemudian aktifkan Jumlah disk Peny impanan. Kolom ini menunjukkan jumlah volume penyimpanan instans.
5. (Opsional) Tambahkan filter ke cakupan lebih lanjut ke jenis instans tertentu yang menarik.

AWS CLI

Untuk mengambil informasi volume penyimpanan instans

Gunakan perintah describe-instance-types. Contoh berikut menampilkan ukuran total penyimpanan instans untuk setiap jenis instans dalam keluarga instans R8i dengan volume penyimpanan instans.

```
aws ec2 describe-instance-types \
    --filters "Name=instance-type,Values=r8i*" "Name=instance-storage-
supported,Values=true" \
    --query 'sort_by(InstanceTypes, &InstanceStorageInfo.TotalSizeInGB)[].
{InstanceType:InstanceType,TotalSizeInGB:InstanceStorageInfo.TotalSizeInGB}' \
    --output table
```

Berikut ini adalah output contoh.

```
-------------------------------------
|        DescribeInstanceTypes       |
+------------------+-----------------+
|   InstanceType   |  TotalSizeInGB  |
+------------------+-----------------+
|  r8id.large      |  118            |
|  r8id.xlarge     |  237            |
|  r8id.2xlarge    |  474            |
|  r8id.4xlarge    |  950            |
|  r8id.8xlarge    |  1900           |
|  r8id.12xlarge   |  2850           |
|  r8id.16xlarge   |  3800           |
|  r8id.24xlarge   |  5700           |
|  r8id.32xlarge   |  7600           |
|  r8id.48xlarge   |  11400          |
|  r8id.metal-48xl |  11400          |
|  r8id.96xlarge   |  22800          |
|  r8id.metal-96xl |  22800          |
+------------------+-----------------+
```

Untuk mendapatkan detail penyimpanan instans lengkap untuk jenis instans

Gunakan perintah describe-instance-types.

```
aws ec2 describe-instance-types \
```

```
    --filters "Name=instance-type,Values=r8id.32xlarge" \
    --query 'InstanceTypes[0].InstanceStorageInfo' \
    --output json
```

Contoh output menunjukkan bahwa jenis instance ini memiliki dua volume SSD NVMe 3800 GB, dengan total penyimpanan instans 7600 GB.

```
{
    "TotalSizeInGB": 7600,
    "Disks": [
        {
            "SizeInGB": 3800,
            "Count": 2,
            "Type": "ssd"
        }
    ],
    "NvmeSupport": "required",
    "EncryptionSupport": "required"
}
```

PowerShell

Untuk mengambil informasi volume penyimpanan instans

Gunakan Get-EC2InstanceType cmdlet. Contoh berikut menampilkan ukuran total penyimpanan instans untuk setiap jenis instans dalam keluarga instans R8i dengan volume penyimpanan instans.

```
(Get-EC2InstanceType -Filter `
    @{Name="instance-type"; Values="r8i*"},
    @{Name="instance-storage-supported"; Values="true"}) |
    Sort-Object {$_.InstanceStorageInfo.TotalSizeInGB} |
    Format-Table InstanceType,

 @{Name="Disks.SizeInGB";Expression={$_.InstanceStorageInfo.Disks[0].SizeInGB}},
        @{Name="Disks.Count";Expression={$_.InstanceStorageInfo.Disks[0].Count}},
        @{Name="TotalSizeInGB";Expression={$_.InstanceStorageInfo.TotalSizeInGB}}
```

Berikut ini adalah output contoh.

```
InstanceType    Disks.SizeInGB Disks.Count TotalSizeInGB
```

```
------------    -------------- ----------- -------------
r8id.large                118           1           118
r8id.xlarge               237           1           237
r8id.2xlarge              474           1           474
r8id.4xlarge              950           1           950
r8id.8xlarge             1900           1          1900
r8id.12xlarge            2850           1          2850
r8id.16xlarge            3800           1          3800
r8id.24xlarge            2850           2          5700
r8id.32xlarge            3800           2          7600
r8id.48xlarge            3800           3         11400
r8id.metal-48xl          3800           3         11400
r8id.96xlarge            3800           6         22800
r8id.metal-96xl          3800           6         22800
```

Untuk mendapatkan detail penyimpanan instans lengkap untuk jenis instans

Gunakan Get-EC2InstanceType cmdlet.

```
(Get-EC2InstanceType `
    -Filter @{Name="instance-type"; Values="r8id.32xlarge"}).InstanceStorageInfo |
    Format-List *,
        @{Name="Disks.Count";Expression={$_.Disks[0].Count}},
        @{Name="Disks.SizeInGB";Expression={$_.Disks[0].SizeInGB}},
        @{Name="Disks.Type";Expression={$_.Disks[0].Type.Value}}
```

Contoh output menunjukkan bahwa jenis instance ini memiliki dua volume SSD NVMe 3800 GB, dengan total penyimpanan instans 7600 GB.

```
Disks             : {Amazon.EC2.Model.DiskInfo}
EncryptionSupport : required
NvmeSupport       : required
TotalSizeInGB     : 7600
Disks.Count       : 2
Disks.SizeInGB    : 3800
Disks.Type        : ssd
```

## Volume penyimpanan instans SSD untuk instans EC2

Seperti volume penyimpanan instans lainnya, Anda harus memetakan volume penyimpanan instans SSD untuk instans Anda saat meluncurkannya. Data di SSD merupakan volume instans SSD

yang hanya bertahan selama masa pakai instans terkait. Untuk informasi selengkapnya, lihat Menambahkan volume penyimpanan instans ke instans EC2.

## Volume SSD NVMe

Beberapa instans menawarkan volume penyimpanan instans solid state drive (SSD) non-volatile memory express (NVMe). Untuk informasi selengkapnya tentang tipe volume penyimpanan instans yang didukung oleh setiap tipe instans, lihat Batas volume penyimpanan instans untuk instans EC2.

Data pada penyimpanan instance NVMe dienkripsi menggunakan sandi XTS-AES-256 blok yang diimplementasikan dalam modul perangkat keras pada instance. Kunci enkripsi dibuat menggunakan modul perangkat keras dan unik untuk setiap perangkat penyimpanan instans NVMe. Semua kunci enkripsi tersebut akan dihancurkan saat instans dihentikan atau diakhiri dan tidak dapat dipulihkan. Anda tidak dapat menonaktifkan enkripsi ini dan Anda tidak dapat menyediakan kunci enkripsi Anda sendiri.

### Instans Linux

Untuk mengakses volume NVMe, driver NVMe harus diinstal. AMI berikut memenuhi persyaratan ini:

- AL2023
- Amazon Linux 2
- Amazon Linux AMI 2018.03 dan setelahnya
- Ubuntu 14.04 atau setelahnya dengan kernel `linux-aws`

> **Note**
>
> AWS Graviton-based jenis instance memerlukan Ubuntu 18.04 atau yang lebih baru dengan kernel `linux-aws`

- Red Hat Enterprise Linux 7.4 atau setelahnya
- SUSE Linux Enterprise Server 12 SP2 atau setelahnya
- CentOS 7.4.1708 atau setelahnya
- FreeBSD 11.1 atau yang lebih baru
- Debian GNU/Linux 9 atau yang lebih baru

- Bottlerocket

Setelah Anda terhubung ke instans, Anda dapat membuat daftar perangkat NVMe menggunakan perintah lspci. Berikut ini adalah contoh output untuk instans `i3.8xlarge`, yang mendukung empat perangkat NVMe.

```
[ec2-user ~]$ lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371SB PIIX3 IDE [Natoma/Triton II]
00:01.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 01)
00:02.0 VGA compatible controller: Cirrus Logic GD 5446
00:03.0 Ethernet controller: Device 1d0f:ec20
00:17.0 Non-Volatile memory controller: Device 1d0f:cd01
00:18.0 Non-Volatile memory controller: Device 1d0f:cd01
00:19.0 Non-Volatile memory controller: Device 1d0f:cd01
00:1a.0 Non-Volatile memory controller: Device 1d0f:cd01
00:1f.0 Unassigned class [ff80]: XenSource, Inc. Xen Platform Device (rev 01)
```

Jika Anda menggunakan sistem operasi yang didukung, tetapi tidak melihat perangkat NVMe, pastikan modul NVMe telah dimuat menggunakan perintah berikut.

- Amazon Linux, Amazon Linux 2, Ubuntu 14/16, Red Hat Enterprise Linux, Server Perusahaan SUSE Linux, CentOS 7

  ```
  $ lsmod | grep nvme
  nvme           48813  0
  ```

- Ubuntu 18

  ```
  $ cat /lib/modules/$(uname -r)/modules.builtin | grep nvme
  s/nvme/host/nvme-core.ko
  kernel/drivers/nvme/host/nvme.ko
  kernel/drivers/nvmem/nvmem_core.ko
  ```

Volume NVMe mematuhi spesifikasi NVMe 1.0e. Anda dapat menggunakan perintah NVMe dengan volume NVMe Anda. Dengan Amazon Linux, Anda dapat menginstal paket `nvme-cli` dari repo menggunakan perintah yum install. Dengan versi Linux lain yang didukung, Anda dapat mengunduh paket `nvme-cli` jika tidak tersedia pada citra.

Instans Windows

AMI AWS Windows terbaru untuk sistem operasi berikut berisi driver AWS NVMe yang digunakan untuk berinteraksi dengan volume penyimpanan instans SSD yang diekspos sebagai perangkat blok NVMe untuk kinerja yang lebih baik:

- Windows Server 2025
- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Windows Server 2012 R2

Setelah Anda terhubung ke instans, Anda dapat memverifikasi bahwa Anda melihat volume NVMe dalam Disk Manager. Pada bilah tugas, buka menu konteks (klik kanan) untuk logo Windows dan pilih Manajemen Disk.

AMI AWS Windows yang disediakan oleh Amazon termasuk driver N AWS VMe. Jika Anda tidak menggunakan AMI AWS Windows terbaru, Anda dapat menginstal driver AWS NVMe saat ini.

## Non-NVMe Volume SSD

Instans berikut mendukung volume penyimpanan instans yang menggunakan SSD non-NVMe untuk memberikan I/O kinerja acak yang tinggi: C3, I2, M3, R3, dan X1. Untuk informasi selengkapnya tentang volume penyimpanan instans yang didukung oleh setiap tipe instans, lihat Batas volume penyimpanan instans untuk instans EC2.

## SSD-based I/O kinerja volume penyimpanan instans

Saat Anda mengisi volume SSD-based penyimpanan instans untuk instance Anda, jumlah IOPS tulis yang dapat Anda capai berkurang. Hal ini disebabkan kerja ekstra yang harus dilakukan pengontrol SSD untuk menemukan ruang yang tersedia, menulis ulang data yang ada, dan menghapus ruang yang tidak digunakan agar dapat ditulis ulang. Proses pengumpulan sampah ini menghasilkan amplifikasi tulis internal ke SSD, yang dinyatakan sebagai rasio operasi tulis SSD terhadap operasi tulis pengguna. Penurunan performa ini bahkan lebih besar jika operasi tulis tidak dalam kelipatan 4.096 byte atau tidak diselaraskan dengan batas 4.096 byte. Jika Anda menulis jumlah byte yang lebih kecil yang tidak selaras, pengontrol SSD harus membaca data di sekitarnya dan menyimpan hasilnya di lokasi baru. Pola ini menghasilkan peningkatan amplifikasi tulis secara signifikan, peningkatan latensi, dan penurunan I/O kinerja secara dramatis.

Pengontrol SSD dapat menggunakan beberapa strategi untuk mengurangi dampak amplifikasi tulis. Salah satu strateginya adalah mencadangkan ruang dalam penyimpanan instans SSD sehingga pengontrol dapat mengelola ruang yang tersedia untuk operasi tulis dengan lebih efisien. Hal ini disebut penyediaan berlebih. Vol SSD-based ume penyimpanan instans yang disediakan untuk instans tidak memiliki ruang yang disediakan untuk penyediaan berlebihan. Untuk mengurangi amplifikasi tulis, sebaiknya biarkan 10 persen volume tidak dipartisi sehingga pengontrol SSD dapat menggunakannya untuk penyediaan berlebihan. Hal ini akan mengurangi penyimpanan yang dapat Anda gunakan, tetapi meningkatkan performa meskipun kapasitas disk hampir penuh.

Misalnya menyimpan volume yang mendukung TRIM, Anda dapat menggunakan perintah TRIM untuk memberi tahu pengontrol SSD kapan pun Anda tidak lagi membutuhkan data yang telah Anda tulis. Hal ini memberikan lebih banyak ruang kosong bagi pengontrol, yang dapat mengurangi amplifikasidan meningkatkan performa. Untuk informasi selengkapnya, lihat Dukungan TRIM volume penyimpanan instans.

## Dukungan TRIM volume penyimpanan instans

Beberapa tipe instans mendukung volume SSD dengan TRIM. Untuk informasi selengkapnya, lihat Batas volume penyimpanan instans untuk instans EC2.

> **Note**
>
> (Hanya instans Windows) Instans yang menjalankan Windows Server 2012 R2 mendukung TRIM sejak Driver AWS PV versi 7.3.0. Instans yang menjalankan versi Windows Server sebelumnya tidak mendukung TRIM.

Volume penyimpanan instans yang mendukung TRIM sepenuhnya dipangkas sebelum dialokasikan ke instans Anda. Volume ini tidak diformat dengan sistem file saat instans diluncurkan, jadi Anda harus memformatnya sebelum dapat dipasang dan digunakan. Untuk akses yang lebih cepat ke volume ini, Anda harus melewati operasi TRIM saat Anda memformatnya.

(Instans Windows) Untuk menonaktifkan sementara dukungan TRIM selama pemformatan awal, gunakan `fsutil behavior set DisableDeleteNotify 1` perintah. Setelah pemformatan selesai, aktifkan kembali dukungan TRIM dengan menggunakan`fsutil behavior set DisableDeleteNotify 0`.

Untuk volume penyimpanan instans yang mendukung TRIM, Anda dapat menggunakan perintah TRIM untuk memberi tahu kontroler SSD setiap kali Anda tidak lagi membutuhkan data yang telah

Anda tulis. Hal ini memberikan lebih banyak ruang kosong bagi kontroler, yang dapat mengurangi amplifikasi dan meningkatkan performa. Pada instance Linux, gunakan `fstrim` perintah untuk mengaktifkan TRIM periodik. Pada instance Windows, gunakan `fsutil behavior set DisableDeleteNotify 0` perintah untuk memastikan dukungan TRIM diaktifkan selama operasi normal.

## Menambahkan volume penyimpanan instans ke instans EC2

Untuk tipe instans dengan volume penyimpanan instans NVMe, semua volume penyimpanan instans yang didukung secara otomatis dilampirkan ke instans saat peluncuran. Volume tersebut secara otomatis dilakukan enumerasi dan diberi nama perangkat saat peluncuran instans.

Misalnya tipe dengan volume penyimpanan instans non-NVMe, seperti C1, C3, M1, M2, M3, R3, D2, H1, I2, X1, dan X1e, Anda harus secara manual menentukan pemetaan perangkat blok untuk volume penyimpanan instans yang ingin Anda lampirkan saat peluncuran. Pemetaan perangkat blok dapat ditentukan dalam permintaan peluncuran instans atau dalam AMI yang digunakan untuk meluncurkan instans. Pemetaan perangkat blok mencakup nama perangkat dan volume yang dipetakannya. Untuk informasi selengkapnya, lihat Memblokir pemetaan perangkat untuk volume pada instans Amazon EC2

> **Important**
>
> Volume penyimpanan instans dapat dilampirkan ke instans hanya saat Anda meluncurkannya. Anda tidak dapat melampirkan volume penyimpanan instans ke instans setelah Anda meluncurkannya.

Setelah Anda meluncurkan suatu instans, Anda harus memastikan bahwa volume penyimpanan instans untuk instans Anda diformat dan dipasang sebelum Anda dapat menggunakannya. Volume root penyimpanan instans dipasang secara otomatis.

Pertimbangan untuk volume root

Pemetaan perangkat blok selalu menentukan volume root untuk instans tersebut. Volume root selalu dipasang secara otomatis.

Instans Linux — Volume root adalah volume Amazon EBS atau volume penyimpanan instans. Untuk instans dengan volume penyimpanan instans untuk volume root, ukuran volume ini bervariasi

berdasarkan AMI, tetapi ukuran maksimumnya adalah 10 GB. Untuk informasi selengkapnya, lihat Tipe volume root.

Instans Windows — Volume root harus berupa volume Amazon EBS. Penyimpanan instans tidak didukung untuk volume root.

Daftar Isi


- Menambahkan volume penyimpanan instans ke AMI Amazon EC2
- Menambahkan volume penyimpanan instans ke instans EC2 selama peluncuran
- Menyediakan volume penyimpanan instans untuk digunakan pada instans EC2


## Menambahkan volume penyimpanan instans ke AMI Amazon EC2

Anda dapat membuat AMI dengan pemetaan perangkat blok yang mencakup volume penyimpanan instans.

Jika Anda meluncurkan instans yang mendukung volume penyimpanan instans non-NVMe menggunakan AMI yang menentukan pemetaan perangkat blok volume penyimpanan instans, instans tersebut menyertakan volume penyimpanan instans. Jika jumlah pemetaan perangkat blok volume penyimpanan instans di AMI melebihi jumlah volume penyimpanan instans yang tersedia untuk instans, pemetaan perangkat blok volume penyimpanan instans tambahan akan diabaikan.

Jika Anda meluncurkan instans yang mendukung volume penyimpanan instans NVMe menggunakan AMI yang menentukan pemetaan perangkat blok volume penyimpanan instans, pemetaan perangkat blok volume penyimpanan instans akan diabaikan. Instans yang mendukung volume penyimpanan instans NVMe mendapatkan semua volume penyimpanan instans yang didukung, terlepas dari pemetaan perangkat blok yang ditentukan dalam permintaan peluncuran instans dan AMI. Pemetaan perangkat volume ini tergantung pada urutan di mana sistem operasi menghitung volume.

Pertimbangan-pertimbangan

- Jumlah volume penyimpanan instans yang tersedia bergantung pada tipe instans. Untuk informasi selengkapnya, lihat the section called “Volume penyimpanan instans yang tersedia”.
- Anda harus menentukan nama perangkat untuk setiap perangkat blok. Untuk informasi selengkapnya, lihat Nama perangkat untuk volume di EC2 instans Amazon.
- Saat Anda meluncurkan instans, Anda dapat menghilangkan volume penyimpanan instans non-NVMe yang ditentukan dalam pemetaan perangkat blok AMI atau menambahkan volume penyimpanan instans.

- Untuk instans M3, tentukan volume penyimpanan instans dalam pemetaan perangkat blok instans, bukan AMI. Amazon EC2 dapat mengabaikan pemetaan perangkat blok volume penyimpanan instans di AMI.

Console

Untuk menambahkan volume penyimpanan instans ke EBS-backed AMI Amazon

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans dan pilih instans.
3. Pilih Tindakan, Citra dan templat, Buat citra.
4. Di halaman Buat citra, masukkan nama dan deskripsi yang berarti untuk citra Anda.
5. Untuk setiap volume penyimpanan instans yang akan ditambahkan, pilih Tambahkan volume, dari Tipe volume pilih volume penyimpanan instan, dan dari Perangkat pilih nama perangkat.
6. Pilih Buat citra.

AWS CLI

Untuk menambahkan volume penyimpanan instans ke AMI

Gunakan perintah create-image dengan `--block-device-mappings` opsi untuk menentukan pemetaan perangkat blok untuk AMI. EBS-backed Gunakan perintah register-image dengan `--block-device-mappings` opsi untuk menentukan pemetaan perangkat blok untuk AMI i S3-backed Amazon.

```
--block-device-mappings file://mapping.json
```

Pemetaan perangkat blok berikut menambahkan dua volume penyimpanan instance.

```
[
    {
        "DeviceName": "/dev/sdc",
        "VirtualName": "ephemeral0"
    },
    {
        "DeviceName": "/dev/sdd",
        "VirtualName": "ephemeral1"
```

```
    }
]
```

PowerShell

Untuk menambahkan volume penyimpanan instans ke AMI

Gunakan New-EC2Image cmdlet dengan `-BlockDeviceMapping` parameter untuk menentukan pemetaan perangkat blok untuk AMI. EBS-backed Gunakan Register-EC2Image cmdlet dengan `-BlockDeviceMapping` parameter untuk menentukan pemetaan perangkat blok untuk Amazon S3-backed AMI.

```
-BlockDeviceMapping $bdm
```

Pemetaan perangkat blok berikut menambahkan dua volume penyimpanan instance.

```
$bdm = @()

$sdc = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$sdc.DeviceName = "/dev/sdc"
$sdc.VirtualName = "ephemeral0"
$bdm += $sdc

$sdd = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$sdd.DeviceName = "/dev/sdd"
$sdd.VirtualName = "ephemeral1"
$bdm += $sdd
```

## Menambahkan volume penyimpanan instans ke instans EC2 selama peluncuran

Saat meluncurkan jenis instans dengan volume penyimpanan instans non-NVMe, seperti C1, C3, M1, M2, M3, R3, D2, H1, I2, X1, dan X1e, Anda harus menentukan pemetaan perangkat blok untuk volume penyimpanan instans yang ingin Anda lampirkan saat peluncuran. Pemetaan perangkat blok harus ditentukan dalam permintaan peluncuran instans atau dalam AMI yang digunakan untuk meluncurkan instans.

Jika AMI menyertakan pemetaan perangkat blok untuk volume penyimpanan instans, Anda tidak perlu menentukan pemetaan perangkat blok dalam permintaan peluncuran instans, kecuali jika Anda membutuhkan lebih banyak volume penyimpanan instans daripada yang disertakan dalam AMI.

Jika AMI tidak menyertakan pemetaan perangkat blok untuk volume penyimpanan instans, Anda harus menentukan pemetaan perangkat blok dalam permintaan peluncuran instans.

Untuk tipe instans dengan volume penyimpanan instans NVMe, semua volume penyimpanan instans yang didukung secara otomatis dilampirkan ke instans saat peluncuran.

Pertimbangan-pertimbangan

- Jumlah volume penyimpanan instans yang tersedia bergantung pada tipe instans. Untuk informasi selengkapnya, lihat the section called “Volume penyimpanan instans yang tersedia”.
- Anda harus menentukan nama perangkat untuk setiap perangkat blok. Untuk informasi selengkapnya, lihat Nama perangkat untuk volume di EC2 instans Amazon.
- Untuk instans M3, Anda dapat menerima volume penyimpanan instans bahkan jika Anda tidak menentukannya dalam pemetaan perangkat blok untuk instans tersebut.

Console

Untuk menentukan pemetaan perangkat blok dalam permintaan peluncuran instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari dasbor, pilih Luncurkan instans.
3. Di bagian Citra Aplikasi dan OS, pilih AMI yang akan digunakan.
4. Di bagian Konfigurasikan penyimpanan, bagian Volume penyimpanan Instans mencantumkan volume penyimpanan instans yang dapat dilampirkan ke instans.
5. Untuk setiap volume penyimpanan instans yang akan dilampirkan, untuk Nama perangkat, pilih nama perangkat yang akan digunakan.
6. Konfigurasikan pengaturan instans yang tersisa sesuai kebutuhan, lalu pilih Luncurkan instans.

AWS CLI

Untuk menentukan pemetaan perangkat blok dalam permintaan peluncuran instance

Gunakan perintah run-instance dengan opsi. `--block-device-mappings`

```
--block-device-mappings file://mapping.json
```

Pemetaan perangkat blok berikut menambahkan dua volume penyimpanan instance.

```
[
    {
        "DeviceName": "/dev/sdc",
        "VirtualName": "ephemeral0"
    },
    {
        "DeviceName": "/dev/sdd",
        "VirtualName": "ephemeral1"
    }
]
```

PowerShell

Untuk menentukan pemetaan perangkat blok dalam permintaan peluncuran instance

Gunakan New-EC2Instance cmdlet dengan opsi. `-BlockDeviceMapping`

```
-BlockDeviceMapping $bdm
```

Pemetaan perangkat blok berikut menambahkan dua volume penyimpanan instance.

```
$bdm = @()

$sdc = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$sdc.DeviceName = "/dev/sdc"
$sdc.VirtualName = "ephemeral0"
$bdm += $sdc

$sdd = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$sdd.DeviceName = "/dev/sdd"
$sdd.VirtualName = "ephemeral1"
$bdm += $sdd
```

## Menyediakan volume penyimpanan instans untuk digunakan pada instans EC2

Setelah meluncurkan instans dengan volume penyimpanan instans terlampir, Anda harus memasang volume sebelum dapat mengaksesnya.

Instans Linux

Anda dapat memformat volume dengan sistem file pilihan Anda setelah meluncurkan instans.

Untuk membuat volume penyimpanan instans tersedia di Linux

1. Sambungkan ke instans Anda menggunakan SSH. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..
2. Gunakan perintah `df -h` untuk melihat volume yang diformat dan dipasang.

```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.8G   72K  3.8G   1% /dev
tmpfs           3.8G     0  3.8G   0% /dev/shm
/dev/nvme0n1p1  7.9G  1.2G  6.6G  15% /
```

3. Gunakan `lsblk` untuk melihat volume yang dipetakan saat peluncuran tetapi tidak diformat dan dipasang.

```
$ lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
nvme0n1       259:1    0    8G  0 disk
##nvme0n1p1   259:2    0    8G  0 part /
##nvme0n1p128 259:3    0    1M  0 part
nvme1n1       259:0    0 69.9G  0 disk
```

4. Untuk memformat dan memasang volume penyimpanan instans yang dipetakan saja, lakukan hal berikut:

   a. Buat sistem file di perangkat menggunakan perintah `mkfs`.

   ```
   $ sudo mkfs -t xfs /dev/nvme1n1
   ```

   b. Buat direktori tempat memasang perangkat menggunakan perintah `mkdir`.

   ```
   $ sudo mkdir /data
   ```

   c. Pasang perangkat di direktori yang baru dibuat dengan menggunakan perintah `mount`.

   ```
   $ sudo mount /dev/nvme1n1 /data
   ```

Instans Windows

Untuk instans Windows, kami memformat ulang volume penyimpanan instans dengan sistem file NTFS.

Anda dapat melihat volume penyimpanan instans menggunakan Manajemen Disk Windows. Untuk informasi selengkapnya, lihat Daftar disk non-NVMe.

Untuk secara manual memasang volume penyimpanan instans

1. Pilih Mulai, masukkan Manajemen Komputer, lalu tekan Enter.
2. Di panel kiri, pilih Manajemen Disk.
3. Jika Anda diminta untuk menginisialisasi volume, pilih volume untuk inisialisasi, pilih jenis partisi yang diperlukan bergantung pada kasus penggunaan Anda, lalu pilih OKE.
4. Dalam daftar volume, buka menu konteks (klik kanan) untuk volume yang akan dipasang, lalu pilih  New Simple Volume.
5. Di wizard, pilih Selanjutnya.
6. Pada layar Tetapkan Ukuran Volume, pilih Selanjutnya untuk menggunakan ukuran volume maksimum. Atau, pilih ukuran volume antara ruang disk minimum dan maksimum.
7. Pada layar Tetapkan Huruf Drive atau Jalur, lakukan salah satu hal berikut ini, lalu pilih Next.
   - Untuk memasang volume dengan huruf drive, pilih Tetapkan huruf drive berikut, lalu pilih huruf drive yang akan digunakan.
   - Untuk memasang volume sebagai folder, pilih Pasang di folder NTFS kosong berikut lalu pilih Jelajahi untuk membuat atau memilih folder yang akan digunakan.
   - Untuk memasang volume tanpa huruf drive atau jalur, pilih Jangan tetapkan huruf drive atau jalur drive.
8. Pada layar Format Partisi, tentukan apakah akan memformat volume. Jika Anda memilih untuk memformat volume, pilih sistem file dan ukuran unit yang diperlukan, dan tentukan label volume.
9. Pilih Selanjutnya, Selesai.

## Aktifkan volume swap penyimpanan instans untuk instans M1 dan C1 EC2

> **Note**
>
> Topik ini berlaku untuk `c1.medium` dan instans `m1.small` Linux saja.

Jenis `c1.medium` dan `m1.small` instance memiliki jumlah memori fisik yang terbatas. Oleh karena itu, mereka diberi volume swap 900 MiB pada saat peluncuran untuk bertindak sebagai memori virtual, atau ruang swap, untuk sistem Linux. Ruang swap di Linux dapat digunakan ketika sistem membutuhkan lebih banyak memori daripada yang dialokasikan secara fisik. Ketika ruang swap diaktifkan, sistem Linux dapat menukar halaman memori yang jarang digunakan dari memori fisik ke ruang swap (baik partisi khusus atau file swap pada sistem file yang ada) dan mengosongkan ruang tersebut untuk halaman memori yang membutuhkan akses berkecepatan tinggi.

> **Note**
>
> - Menggunakan ruang swap untuk paging memori tidak secepat atau seefisien menggunakan RAM. Jika beban kerja Anda secara teratur memindahkan memori ke ruang swap, Anda harus mempertimbangkan untuk bermigrasi ke tipe instans yang lebih besar dengan RAM yang lebih besar. Untuk informasi selengkapnya, lihat Perubahan jenis instans Amazon EC2.
> - Meskipun kernel Linux melihat ruang swap ini sebagai partisi pada volume root, sebenarnya itu adalah volume penyimpanan instance terpisah, terlepas dari jenis volume root Anda.

Amazon Linux secara otomatis mengaktifkan dan menggunakan ruang swap ini, tetapi AMI Anda mungkin memerlukan beberapa langkah tambahan untuk mengenali dan menggunakan ruang swap ini. Untuk melihat apakah instans Anda menggunakan ruang swap, Anda dapat menggunakan perintah swapon -s.

```
[ec2-user ~]$ swapon -s
Filename                                Type            Size    Used    Priority
/dev/xvda3                              partition       917500  0       -1
```

Instans di atas memiliki 900 MiB volume swap yang dilampirkan dan diaktifkan. Jika Anda tidak melihat volume swap yang tercantum dalam perintah ini, Anda mungkin perlu mengaktifkan ruang swap untuk perangkat. Periksa disk yang tersedia menggunakan perintah lsblk.

```
[ec2-user ~]$ lsblk
NAME  MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda1 202:1    0    8G  0 disk /
xvda3 202:3    0  896M  0 disk
```

Di sini, volume swap `xvda3` tersedia untuk instans, tetapi tidak diaktifkan (perhatikan bahwa `MOUNTPOINT` bidang kosong). Anda dapat mengaktifkan volume swap dengan perintah swapon.

> **Note**
>
> Anda harus melakukan prepend `/dev/` ke nama perangkat yang terdaftar oleh lsblk. Perangkat Anda mungkin diberi nama berbeda, seperti `sda3`, `sde3`, atau `xvde3`. Gunakan nama perangkat untuk sistem Anda pada perintah di bawah ini.

```
[ec2-user ~]$ sudo swapon /dev/xvda3
```

Sekarang, ruang swap akan muncul dalam output lsblk dan swapon -s.

```
[ec2-user ~]$ lsblk
NAME  MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda1 202:1    0    8G  0 disk /
xvda3 202:3    0  896M  0 disk [SWAP]
[ec2-user ~]$ swapon -s
Filename                                Type            Size    Used    Priority
/dev/xvda3                              partition       917500  0       -1
```

Anda juga harus mengedit file `/etc/fstab` sehingga ruang swap ini secara otomatis diaktifkan di setiap boot sistem.

```
[ec2-user ~]$ sudo vim /etc/fstab
```

Tambahkan baris berikut ke file `/etc/fstab` (menggunakan nama perangkat swap untuk sistem Anda):

```
/dev/xvda3       none    swap    sw  0       0
```

Untuk menggunakan volume penyimpanan instans sebagai ruang swap

Setiap volume penyimpanan instans dapat digunakan sebagai ruang swap. Misalnya, tipe instans `m3.medium` mencakup volume penyimpanan instans SSD 4 GB yang sesuai untuk ruang swap. Jika volume penyimpanan instans Anda jauh lebih besar (misalnya, 350 GB), Anda dapat mempertimbangkan untuk membagi volume tersebut dengan partisi pertukaran yang lebih kecil sebesar 4-8 GB dan sisanya untuk volume data.

> **Note**
>
> Prosedur ini hanya berlaku untuk tipe instans yang mendukung penyimpanan instans. Untuk daftar tipe instans yang didukung, lihat Batas volume penyimpanan instans untuk instans EC2.

1. Buat daftar perangkat blok yang dilampirkan ke instans Anda untuk mendapatkan nama perangkat untuk volume penyimpanan instans Anda.

```
[ec2-user ~]$ lsblk -p
NAME       MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
/dev/xvdb  202:16   0   4G  0 disk /media/ephemeral0
/dev/xvda1 202:1    0   8G  0 disk /
```

   Dalam contoh ini, volume penyimpanan instans adalah `/dev/xvdb`. Karena ini adalah instans Amazon Linux, volume penyimpanan instans diformat dan dipasang di `/media/ephemeral0`; tidak semua sistem operasi Linux melakukan ini secara otomatis.

2. (Opsional) Jika volume penyimpanan instans Anda dipasang (mencantumkan `MOUNTPOINT` dalam output perintah lsblk), lepaskan dengan perintah berikut ini.

```
[ec2-user ~]$ sudo umount /dev/xvdb
```

3. Siapkan area swap Linux di perangkat dengan perintah mkswap.

```
[ec2-user ~]$ sudo mkswap /dev/xvdb
mkswap: /dev/xvdb: warning: wiping old ext3 signature.
Setting up swapspace version 1, size = 4188668 KiB
no label, UUID=b4f63d28-67ed-46f0-b5e5-6928319e620b
```

4. Aktifkan ruang swap baru.

```
[ec2-user ~]$ sudo swapon /dev/xvdb
```

5. Verifikasi bahwa ruang swap baru sedang digunakan.

```
[ec2-user ~]$ swapon -s
Filename    Type  Size Used Priority
/dev/xvdb                               partition 4188668 0 -1
```

6. Edit file `/etc/fstab` sehingga ruang swap ini secara otomatis diaktifkan di setiap boot sistem.

```
[ec2-user ~]$ sudo vim /etc/fstab
```

Jika file `/etc/fstab` memiliki entri untuk `/dev/xvdb` (atau `/dev/sdb`), ubah agar cocok dengan baris di bawah ini; jika tidak memiliki entri untuk perangkat ini, tambahkan baris berikut ke file `/etc/fstab` (menggunakan nama perangkat swap untuk sistem Anda):

```
/dev/xvdb       none    swap    sw  0       0
```

> **Important**
>
> Data volume penyimpanan instans hilang saat instans dihentikan atau dihibernasi; ini termasuk pemformatan ruang swap penyimpanan instans yang dibuat dalam file Step 3. Jika Anda berhenti dan memulai ulang suatu instans yang telah dikonfigurasi untuk menggunakan ruang swap penyimpanan instans, Anda harus mengulangi Step 1 melalui Step 5 pada volume penyimpanan instan baru.

## Inisialisasi volume penyimpanan instans pada instans EC2

Karena cara Amazon EC2 dalam memvirtualisasi disk, penulisan pertama ke lokasi mana pun pada beberapa volume penyimpanan instans berjalan lebih lambat dibandingkan penulisan berikutnya. Untuk sebagian besar aplikasi, mengamortisasi biaya ini selama masa pakai instans dapat diterima. Namun, jika Anda membutuhkan performa disk yang tinggi, sebaiknya inisialisasi drive dengan menulis satu kali ke setiap lokasi drive sebelum digunakan dalam produksi.

> **Note**
>
> Jenis instans dengan solid state drive (SSD) yang terpasang langsung dan dukungan TRIM memberikan kinerja maksimum pada waktu peluncuran, tanpa inisialisasi. Untuk informasi tentang penyimpanan instans untuk setiap tipe instans, lihat Batas volume penyimpanan instans untuk instans EC2.

Jika Anda membutuhkan fleksibilitas yang lebih besar dalam hal latensi atau throughput, sebaiknya gunakan Amazon EBS.

Untuk menginisialisasi volume penyimpanan instans, gunakan perintah dd berikut, bergantung pada penyimpanan yang akan menginisialisasi (misalnya, `/dev/sdb` atau `/dev/nvme1n1`).

> **Note**
>
> Pastikan untuk melepaskan drive sebelum melakukan perintah ini.
> Inisialisasi dapat memakan waktu lama (sekitar 8 jam untuk instans sangat besar).

Untuk menginisialisasi volume penyimpanan instans, gunakan perintah berikut pada tipe instans `m1.large`, `m1.xlarge`, `c1.xlarge`, `m2.xlarge`, `m2.2xlarge`, dan `m2.4xlarge`:

```
dd if=/dev/zero of=/dev/sdb bs=1M
dd if=/dev/zero of=/dev/sdc bs=1M
dd if=/dev/zero of=/dev/sdd bs=1M
dd if=/dev/zero of=/dev/sde bs=1M
```

Untuk melakukan inisialisasi pada semua volume penyimpanan instans pada saat bersamaan, gunakan perintah berikut:

```
dd if=/dev/zero bs=1M|tee /dev/sdb|tee /dev/sdc|tee /dev/sde > /dev/sdd
```

Mengonfigurasi drive untuk RAIDdengan melakukanke setiap lokasi drive. Saat mengonfigurasi RAID berbasis perangkat lunak, pastikan untuk mengubah kecepatan rekonstruksi minimum:

```
echo $((30*1024)) > /proc/sys/dev/raid/speed_limit_min
```

## Statistik kinerja terperinci untuk volume penyimpanan instans Amazon EC2

Amazon EC2 menyediakan statistik kinerja resolusi tinggi real-time untuk volume penyimpanan instans NVMe yang terpasang pada instans Nitro-based Amazon EC2. Statistik ini disajikan sebagai penghitung agregat yang disimpan selama masa pakai instance. Statistik memberikan rincian tentang jumlah operasi kumulatif, byte yang dikirim dan diterima, waktu yang dihabiskan untuk I/O operasi baca dan tulis, dan histogram untuk operasi baca dan tulis I/O . Meskipun statistik ini mempertahankan konsistensi dengan statistik kinerja terperinci Amazon EBS, statistik ini juga menyertakan histogram latensi terperinci yang dipecah berdasarkan I/O ukuran, yang dapat memberikan wawasan yang lebih terperinci tentang pola kinerja penyimpanan Anda. Visibilitas yang ditingkatkan ini memungkinkan Anda mengidentifikasi secara tepat I/O ukuran tertentu mana

yang mengalami masalah latensi, memungkinkan Anda untuk mengoptimalkan kinerja aplikasi dan memecahkan masalah dengan lebih efektif.

Anda dapat mengumpulkan statistik ini dengan perincian hingga interval 1 detik. Jika permintaan dibuat lebih sering dari interval 1 detik, driver NVMe mungkin mengantri permintaan, bersama dengan perintah admin lainnya, untuk diproses di lain waktu.

Pertimbangan-pertimbangan

- Statistik hanya didukung untuk volume penyimpanan instans NVMe yang dilampirkan ke Nitro-based instans.
- Penghitung tidak persisten di seluruh penghentian instans dan restart.
- Statistik tersedia tanpa biaya tambahan.

## Statistik

Perangkat blok NVMe menjual statistik berikut:

| Nama statistik | Nama lengkap | Tipe | Deskripsi |
|---|---|---|---|
| `total_rea d_ops` | Total operasi baca | Penghitun g | Jumlah total operasi baca yang selesai. |
| `total_wr te_ops` | Total operasi penulisan | Penghitun g | Jumlah total operasi penulisan yang diselesai kan. |
| `total_rea d_bytes` | Jumlah byte yang dibaca | Penghitun g | Jumlah total byte baca yang ditransfer. |
| `total_wr te_bytes` | Jumlah byte tulis | Penghitun g | Jumlah total byte tulis yang ditransfer. |
| `total_rea d_time` | Total waktu baca | Penghitun g | Total waktu yang dihabiskan, dalam mikrodetik, oleh semua operasi baca yang selesai. |
| `total_wr te_time` | Total waktu penulisan | Penghitun g | Total waktu yang dihabiskan, dalam mikrodeti k, oleh semua operasi penulisan yang diselesai kan. |

| Nama statistik | Nama lengkap | Tipe | Deskripsi |
|---|---|---|---|
| instance_store_volume_performance_exceeded_iops | Total permintaan waktu melebihi IOPS maksimum volume | Penghitung | Total waktu, dalam mikrodetik, permintaan IOPS melebihi IOPS maksimum volume. Nilai apa pun di atas 0 menunjukkan bahwa beban kerja Anda menuntut lebih banyak IOPS daripada volume yang dapat diberikan. Idealnya, jumlah tambahan metrik ini, antara dua waktu snapshot, harus minimal. |
| instance_store_volume_performance_exceeded_tp | Total permintaan waktu melebihi throughput maksimum volume | Penghitung | Total waktu, dalam mikrodetik, permintaan throughput melebihi throughput maksimum volume. Nilai apa pun di atas 0 menunjukkan bahwa beban kerja Anda menuntut lebih banyak throughput daripada volume yang dapat diberikan. Idealnya, jumlah tambahan metrik ini, antara dua waktu snapshot, harus minimal. |
| volume_queue_length | Panjang antrian volume | Titik waktu | Jumlah operasi baca dan tulis yang menunggu untuk diselesaikan. |
| read_io_latency_histogram | Baca I/O histogram | Histogram * | Jumlah operasi baca yang diselesaikan dalam setiap bin latensi, dalam mikrodetik. |
| write_io_latency_histogram | Tulis I/O histogram | Histogram * | Jumlah operasi tulis yang diselesaikan dalam setiap bin latensi, dalam mikrodetik. |

> **Note**
>
> * Statistik histogram hanya mewakili I/O operasi yang telah selesai dengan sukses. I/O Operasi yang terhenti atau terganggu tidak termasuk, tetapi akan terlihat dalam `volume_queue_length` statistik, yang disajikan sebagai statistik point-in-time.

## Mengakses statistik

Statistik harus diakses langsung dari instance yang dilampirkan volume penyimpanan instans. Anda dapat mengakses statistik menggunakan salah satu metode berikut.

Instans Linux

Amazon CloudWatch

Anda dapat mengonfigurasi CloudWatch agen Amazon untuk mengumpulkan statistik dari instans Anda dan membuatnya tersedia sebagai metrik khusus di CloudWatch. Anda kemudian dapat menggunakan metrik CloudWatch untuk menganalisis I/O pola, melacak tren kinerja, membuat dasbor khusus, dan mengatur alarm otomatis berdasarkan ambang kinerja.

Untuk informasi selengkapnya tentang mengonfigurasi CloudWatch agen, lihat Meng umpulkan metrik volume penyimpanan instans Amazon EC2.

nvme-cli tool

Untuk mengakses statistik

1. Hubungkan ke instance tempat volume dilampirkan.
2. AMI Amazon Linux 2023 yang dirilis setelah 15 September 2025 menyertakan versi terbaru dari `nvme-cli` alat ini. Jika Anda menggunakan AMI Amazon Linux yang lebih lama, perbarui `nvme-cli` alat ini.

   ```
   sudo yum install nvme-cli
   ```

3. Jalankan perintah berikut dan tentukan nama perangkat untuk volume.

   ```
   sudo nvme amzn stats /dev/nvme0n1
   ```

Statistik juga memberikan histogram latensi terperinci yang dipecah berdasarkan I/O ukuran. Untuk melihat statistik yang dipecah berdasarkan I/O ukuran, sertakan `--details` opsi. Contoh:

```
sudo nvme amzn stats --details /dev/nvme0n1
```

Anda bisa mendapatkan informasi lebih lanjut tentang cara menggunakan alat ini dengan menentukan `--help` opsi. Contoh:

```
sudo nvme amzn stats --help
```

Instans Windows

nvme_amzn.exe tool

Untuk mengakses statistik

1. Hubungkan ke instance tempat volume dilampirkan.
2. Pastikan Anda menggunakan versi `1.7.0` driver AwsnVMe atau yang lebih baru. Untuk informasi selengkapnya tentang memperbarui driver AWSNVMe, lihat AWS driver NVMe.
3. Dapatkan nomor disk untuk volume. Untuk informasi selengkapnya, lihat Mem etakan disk NVMe pada instans Windows Amazon EC2 ke volume.
4. Jalankan perintah berikut sebagai Administrator dan tentukan nomor disk untuk volume.

   ```
   .\nvme_amzn.exe stats disk_number
   ```

Statistik juga memberikan histogram latensi terperinci yang dipecah berdasarkan I/O ukuran. Untuk melihat statistik yang dipecah berdasarkan I/O ukuran, sertakan `--details` opsi. Contoh:

```
.\nvme_amzn.exe stats --details disk_number
```

# Volume root untuk instans Amazon EC2 Anda

Saat Anda meluncurkan instans, kami membuat volume root untuk instans tersebut. Volume root berisi gambar yang digunakan untuk booting instans tersebut. Setiap instans memiliki volume root tunggal. Anda dapat menambahkan volume penyimpanan ke instans selama atau setelah peluncuran.

AMI yang Anda gunakan untuk meluncurkan instance menentukan jenis volume root. Anda dapat meluncurkan instance dari Amazon EBS-backed AMI (instans Linux dan Windows) atau Amazon S3-backed AMI (hanya instans Linux). Ada perbedaan yang signifikan antara apa yang dapat Anda lakukan dengan setiap jenis AMI. Untuk informasi selengkapnya tentang metrik ini, lihat Tipe volume root.

Sebaiknya gunakan AMI yang didukung oleh Amazon EBS, karena instans ini diluncurkan lebih cepat dan menggunakan penyimpanan persisten.

Kami mencadangkan nama perangkat khusus untuk volume root. Untuk informasi selengkapnya, lihat Nama perangkat untuk volume di EC2 instans Amazon.

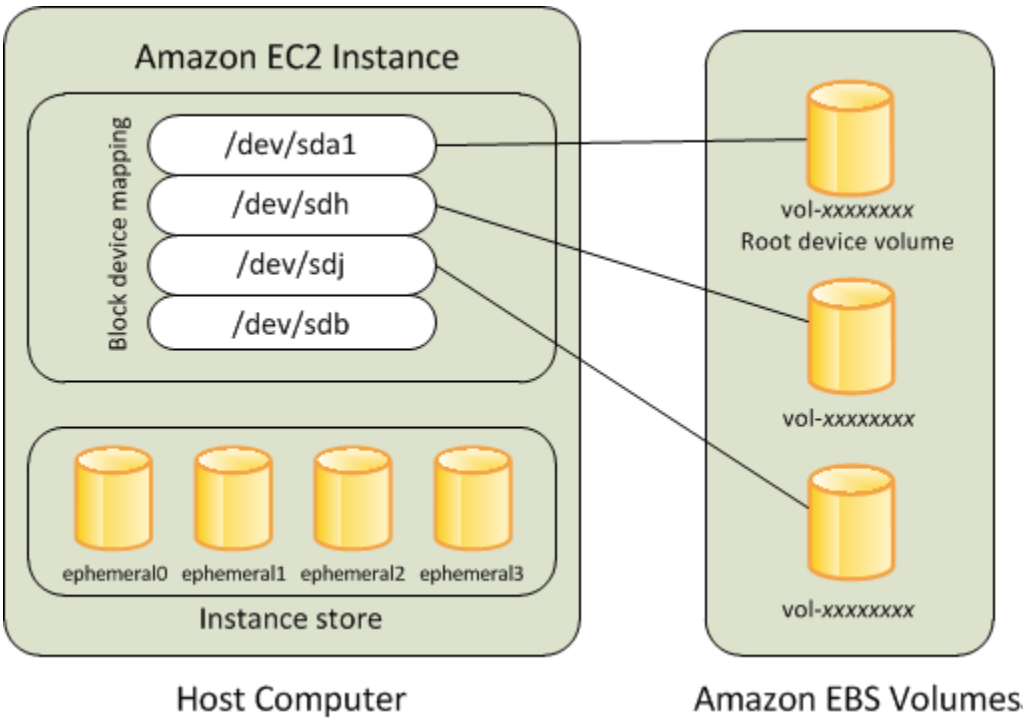
Daftar Isi


- Instans dengan volume root Amazon EBS
- Instans dengan instance menyimpan volume root (hanya instans Linux)
- Menyimpan volume root Amazon EBS setelah instans Amazon EC2 berakhir
- Ganti volume root untuk instans Amazon EC2 tanpa menghentikannya


## Instans dengan volume root Amazon EBS

Instans yang menggunakan Amazon EBS untuk volume root secara otomatis memiliki volume Amazon EBS yang terlampir. Saat Anda meluncurkan instance dengan Amazon EBS-AMI, kami membuat volume Amazon EBS untuk setiap snapshot Amazon EBS yang direferensikan oleh AMI. Anda dapat menggunakan volume Amazon EBS lain atau volume penyimpanan instans, bergantung pada tipe instans.

Instans dengan volume root EBS dapat dihentikan dan kemudian dimulai ulang tanpa mempengaruhi data yang disimpan dalam volume terlampir. Ada berbagai tugas terkait instans dan volume yang dapat Anda lakukan ketika instance dengan volume root EBS berada dalam keadaan berhenti. Misalnya, Anda dapat memodifikasi properti instance, mengubah ukurannya, atau memperbarui kernel yang digunakannya, atau Anda dapat melampirkan volume root ke instance berjalan yang berbeda untuk debugging atau tujuan lainnya. Untuk informasi selengkapnya, lihat volume https://docs.aws.amazon.com/ebs/latest/userguide/ebs-volumes.html Amazon EBS.




Batasan

Anda tidak dapat menggunakan volume EBS `st1` atau `sc1` sebagai volume root.

Kegagalan instans

Jika instance dengan volume root EBS gagal, Anda dapat memulihkan sesi dengan mengikuti salah satu metode berikut:

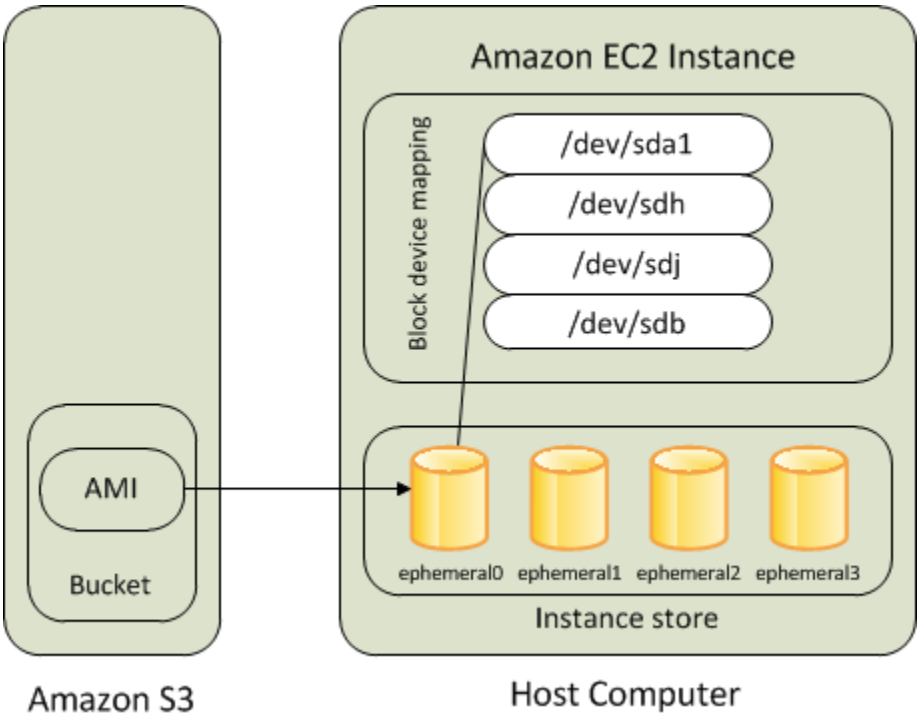
- Hentikan dan mulai lagi (coba metode ini terlebih dahulu).
- Ambil semua volume yang relevan dan buat AMI baru secara otomatis. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.
- Lampirkan volume ke instans baru dengan mengikuti langkah-langkah ini:
  1. Buat snapshot dari volume root.
  2. Daftarkan AMI baru menggunakan snapshot.
  3. Luncurkan instans baru dari AMI baru.
  4. Lepaskan volume Amazon EBS yang tersisa dari instans lama.
  5. Lampirkan kembali volume Amazon EBS ke instans baru.

# Instans dengan instance menyimpan volume root (hanya instans Linux)

> **Note**
>
> Instans Windows tidak mendukung volume root penyimpanan instance.

Instans yang menggunakan penyimpanan instans untuk volume root secara otomatis memiliki satu atau lebih volume penyimpanan instans yang tersedia, dengan satu volume berfungsi sebagai volume root. Ketika sebuah instance diluncurkan dari Amazon S3-backed AMI, AMI disalin ke volume root. Perhatikan bahwa Anda dapat menggunakan volume penyimpanan instans tambahan, bergantung pada tipe instans.

Setiap data pada volume penyimpanan instans tetap ada selama instans berjalan, tetapi data ini dihapus saat instance dihentikan (instans dengan volume root penyimpanan instans tidak mendukung tindakan Stop) atau jika gagal (seperti jika drive yang mendasarinya mengalami masalah). Untuk informasi selengkapnya, lihat Penyimpanan blok sementara instans untuk instans EC2.




Tipe instans yang didukung

Hanya jenis instans berikut yang mendukung volume penyimpanan instans sebagai volume root: C1, C3, D2, I2, M1, M2, M3, R3, dan X1.

Kegagalan Instans

Setelah instance dengan volume root penyimpanan instance gagal atau berakhir, instans tidak dapat dipulihkan. Jika Anda berencana menggunakan instans dengan volume root penyimpanan instans, sebaiknya Anda mendistribusikan data pada volume penyimpanan instans di beberapa Zona Ketersediaan. Anda juga harus mencadangkan data penting dari volume penyimpanan instans Anda ke penyimpanan persisten secara teratur.

## Menyimpan volume root Amazon EBS setelah instans Amazon EC2 berakhir

Secara default, volume root Amazon EBS untuk instans dihapus saat instans dihentikan. Anda dapat mengubah perilaku default untuk memastikan bahwa volume root Amazon EBS tetap ada setelah instans dihentikan. Untuk mengubah perilaku default, atur `DeleteOnTermination` atribut `kefalse`. Anda dapat melakukannya baik saat peluncuran instance atau nanti.

Tugas

- Mengonfigurasi volume root agar tetap ada selama peluncuran instans
- Konfigurasikan volume root agar tetap ada untuk instans yang ada
- Konfirmasikan bahwa volume root dikonfigurasi agar tetap ada

### Mengonfigurasi volume root agar tetap ada selama peluncuran instans

Anda dapat mengonfigurasi volume root agar tetap ada saat Anda meluncurkan instance.

Console

Untuk mengonfigurasi volume root agar tetap ada saat Anda meluncurkan instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans lalu pilih Luncurkan instans.
3. Pilih Amazon Machine Image (AMI), pilih tipe instans, pilih pasangan kunci, dan konfigurasikan pengaturan jaringan Anda.
4. Untuk Konfigurasikan penyimpanan, pilih Lanjutan.
5. Perluas volume root.
6. Untuk Hapus saat pengakhiran, pilih Tidak.
7. Setelah Anda selesai mengonfigurasi instans, pilih Luncurkan instans.

AWS CLI

Untuk mengonfigurasi volume root agar tetap ada saat Anda meluncurkan instance

Gunakan perintah run-instance dan sertakan opsi berikut.

```
--block-device-mappings file://mapping.json
```

Di`mapping.json`, tentukan pemetaan perangkat blok yang menetapkan `DeleteOnTermination` atribut ke`false`.

```
[
    {
        "DeviceName": "/dev/sda1",
        "Ebs": {
            "DeleteOnTermination": false
        }
    }
]
```

PowerShell

Untuk mengonfigurasi volume root agar tetap ada saat Anda meluncurkan instance

Gunakan New-EC2Instance cmdlet dan sertakan parameter berikut.

```
-BlockDeviceMapping $bdm
```

Buat pemetaan perangkat blok yang menetapkan `DeleteOnTermination` atribut ke`$false`.

```
$ebs = New-Object Amazon.EC2.Model.EbsBlockDevice
$ebs.DeleteOnTermination = $false
$bdm = New-Object Amazon.EC2.Model.BlockDeviceMapping
$bdm.DeviceName = "dev/xvda"
$bdm.Ebs = $ebs
```

## Konfigurasikan volume root agar tetap ada untuk instans yang ada

Anda dapat mengonfigurasi volume root untuk bertahan untuk instance yang sedang berjalan. Perhatikan bahwa Anda tidak dapat menyelesaikan tugas ini menggunakan konsol Amazon EC2.

AWS CLI

Untuk mengonfigurasi volume root agar tetap ada untuk instance yang ada

Gunakan perintah modify-instance-attribute dengan pemetaan perangkat blok yang menetapkan atribut `DeleteOnTermination` ke `false`.

```
aws ec2 modify-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --block-device-mappings file://mapping.json
```

Tentukan hal berikut dalam `mapping.json`.

```
[
    {
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "DeleteOnTermination": false
        }
    }
]
```

PowerShell

Untuk mengonfigurasi volume root agar tetap ada untuk instance yang ada

Gunakan Edit-EC2InstanceAttribute cmdlet dengan pemetaan perangkat blok yang menetapkan `DeleteOnTermination` atribut ke. `$false`

```
$ebs = New-Object Amazon.EC2.Model.EbsInstanceBlockDeviceSpecification
$ebs.DeleteOnTermination = $false
$bdm = New-Object Amazon.EC2.Model.InstanceBlockDeviceMappingSpecification
$bdm.DeviceName = "/dev/xvda"
$bdm.Ebs = $ebs
Edit-EC2InstanceAttribute `
    -InstanceId i-1234567890abcdef0 `
    -BlockDeviceMapping $bdm
```

## Konfirmasikan bahwa volume root dikonfigurasi agar tetap ada

Anda dapat mengonfirmasi bahwa volume root dikonfigurasi untuk bertahan.

Console

Untuk mengonfirmasi bahwa volume root dikonfigurasi untuk bertahan

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans kemudian pilih instans Anda.
3. Di tab Penyimpanan, pada Perangkat blok, cari entri untuk volume root. Jika Hapus saat pengakhiran adalah No, volume dikonfigurasi untuk dipertahankan.

AWS CLI

Untuk mengonfirmasi bahwa volume root dikonfigurasi untuk bertahan

Gunakan perintah describe-instance dan verifikasi bahwa `DeleteOnTermination` atribut disetel ke. `false`

```
aws ec2 describe-instances \
    --instance-id i-1234567890abcdef0 \
    --query "Reservations[].Instances[].BlockDeviceMappings"
```

Berikut ini adalah output contoh.

```
[
    [
        {
            "DeviceName": "/dev/xvda",
            "Ebs": {
                "AttachTime": "2024-07-12T04:05:33.000Z",
                "DeleteOnTermination": false,
                "Status": "attached",
                "VolumeId": "vol-1234567890abcdef0"

        }
    ]
]
```

PowerShell

Untuk mengonfirmasi bahwa volume root dikonfigurasi untuk bertahan

Gunakan Get-EC2Instance cmdlet dan verifikasi bahwa `DeleteOnTermination` atribut disetel ke. `False`

```
(Get-EC2Instance -InstanceId i-
i-1234567890abcdef0).Instances.BlockDeviceMappings.Ebs
```

Berikut ini adalah output contoh.

```
AssociatedResource  :
AttachTime          : 7/12/2024 4:05:33 AM
DeleteOnTermination : False
Operator            :
Status              : attached
VolumeId            : vol-1234567890abcdef0
```

## Ganti volume root untuk instans Amazon EC2 tanpa menghentikannya

Amazon EC2 memungkinkan Anda untuk mengganti volume root Amazon EBS untuk instans yang berjalan sambil mempertahankan hal berikut:

- Data yang disimpan di volume penyimpanan instans — Volume penyimpanan instans tetap dilampirkan ke instans setelah volume root dipulihkan.
- Data yang disimpan pada volume Amazon EBS data (non-root) — Volume Non-root Amazon EBS tetap melekat pada instans setelah volume root dipulihkan.
- Konfigurasi jaringan — Semua antarmuka jaringan tetap dilampirkan ke instans dan mempertahankan alamat IP, pengidentifikasi, dan ID lampiran. Ketika instans menjadi tersedia, semua lalu lintas jaringan yang tertunda dibersihkan. Selain itu, instans tetap pada host fisik yang sama, sehingga mempertahankan alamat IP publik dan privat serta nama DNS.
- Kebijakan IAM — profil dan kebijakan IAM (seperti kebijakan berbasis tanda) yang terkait dengan instans dipertahankan dan diberlakukan.

Daftar Isi


- Cara kerja penggantian volume root
- Pertimbangan-pertimbangan
- Mengganti volume root

## Cara kerja penggantian volume root

Saat Anda mengganti volume root untuk sebuah instance, kami membuat tugas penggantian volume  root. Volume root asli dilepaskan dari instans, dan volume root baru dilampirkan ke instans sebagai gantinya. Pemetaan perangkat blok instans diperbarui untuk mencerminkan ID volume root pengganti.

Saat mengganti volume root untuk sebuah instance, Anda harus menentukan sumber untuk volume baru. Berikut ini adalah opsi yang mungkin.

Kembalikan volume root ke keadaan semula

Opsi ini menggantikan volume root saat ini dengan volume yang didasarkan pada snapshot yang digunakan untuk membuatnya.

Pertimbangan untuk menggunakan status peluncuran

Volume root pengganti mendapatkan tipe, ukuran, dan atribut pengakhiran yang sama dengan volume root asli.

Ganti volume root menggunakan snapshot

Opsi ini menggantikan volume root saat ini dengan volume pengganti yang didasarkan pada snapshot yang Anda tentukan. Misalnya, snapshot tertentu yang sebelumnya Anda buat dari volume root ini. Ini berguna jika Anda perlu memulihkan dari masalah yang disebabkan oleh kerusakan volume root atau kesalahan konfigurasi jaringan di sistem operasi tamu.

Volume root pengganti mendapatkan tipe, ukuran, dan atribut pengakhiran yang sama dengan volume root asli.

Pertimbangan untuk menggunakan snapshot

- Anda hanya dapat menggunakan snapshot yang dibuat langsung dari volume root instans saat ini atau sebelumnya.
- Anda tidak dapat menggunakan salinan snapshot yang dibuat dari snapshot yang diambil dari volume root.
- Setelah berhasil mengganti volume root, Anda masih dapat menggunakan snapshot yang diambil dari volume root asli untuk mengganti volume root (pengganti) yang baru.

Ganti volume root menggunakan AMI

Opsi ini menggantikan volume root saat ini menggunakan AMI yang Anda tentukan. Ini berguna jika Anda perlu melakukan penambalan atau peningkatan sistem operasi dan aplikasi. AMI harus memiliki kode produk, informasi penagihan, jenis arsitektur, dan jenis virtualisasi yang sama dengan instance.

Jika instance diaktifkan untuk ENA atau sriov-net, maka Anda harus menggunakan AMI yang mendukung fitur-fitur tersebut. Jika instance tidak diaktifkan untuk ENA atau sriov-net, maka Anda dapat memilih AMI yang tidak menyertakan dukungan untuk fitur-fitur tersebut, atau Anda dapat menambahkan dukungan secara otomatis jika Anda memilih AMI yang mendukung ENA atau sriov-net.

Jika instance diaktifkan untuk NitroTPM, maka Anda harus menggunakan AMI yang mengaktifkan NitroTPM. Dukungan NitroTPM tidak diaktifkan jika instance tidak dikonfigurasi untuk itu, terlepas dari AMI yang Anda pilih.

Anda dapat memilih AMI dengan mode boot yang berbeda dari instans, selama instans mendukung mode boot AMI. Jika instans tidak mendukung mode boot, permintaan gagal. Jika instans mendukung mode boot, mode boot baru disebarkan ke instans dan data UEFI-nya diperbarui sesuai dengan itu. Jika Anda memodifikasi urutan boot secara manual atau menambahkan kunci Boot Aman UEFI pribadi untuk memuat modul kernel pribadi, perubahan akan hilang selama penggantian volume root.

Volume root pengganti mendapatkan jenis volume yang sama dan menghapus pada atribut terminasi seperti volume root asli. Ukuran volume root pengganti adalah ukuran pemetaan perangkat blok volume root AMI yang lebih besar dan ukuran volume root saat ini.

Setelah tugas penggantian volume root selesai, informasi baru dan terbaru berikut ini tercermin saat Anda menjelaskan instance menggunakan konsol, AWS CLI atau AWS SDK:

- ID AMI Baru
- ID volume baru untuk volume root
- Konfigurasi mode boot yang diperbarui (jika diubah oleh AMI)
- Konfigurasi NitroTPM yang diperbarui (jika diaktifkan oleh AMI)
- Konfigurasi ENA yang diperbarui (jika diaktifkan oleh AMI)
- Konfigurasi sriov-net yang diperbarui (jika diaktifkan oleh AMI)

ID AMI baru juga tercermin dalam metadata instans.

Pertimbangan untuk menggunakan AMI:

- Jika Anda menggunakan AMI yang memiliki beberapa pemetaan perangkat blok, hanya volume root AMI yang digunakan. Volume lainnya (non-root) diabaikan.
- Anda hanya dapat menggunakan fitur ini jika Anda memiliki izin untuk AMI dan snapshot volume root terkait. Anda tidak dapat menggunakan fitur ini dengan AWS Marketplace AMI.
- Anda hanya dapat menggunakan AMI tanpa kode produk hanya jika instans tidak memiliki kode produk.
- Ukuran volume root pengganti adalah ukuran pemetaan perangkat blok volume root AMI yang lebih besar dan ukuran volume root saat ini.
- Dokumen identitas instans untuk instans diperbarui secara otomatis.
- Jika instans mendukung NitroTPM, data NitroTPM untuk instans direset dan kunci baru dihasilkan.

Ganti volume root menggunakan volume Amazon EBS yang ada

Berbeda dengan opsi snapshot dan AMI, yang membuat volume baru, opsi ini menggantikan volume root saat ini dengan volume Amazon EBS yang sudah Anda konfigurasi. Ini berguna untuk beban kerja stateful, seperti database, di mana Anda ingin menyiapkan metadata atau perangkat lunak pada volume root sebelum Anda melampirkannya ke instance.

Sebelum menggunakan opsi ini, Anda harus membuat dan menyiapkan volume Amazon EBS pengganti. Anda dapat menyiapkannya dengan melampirkannya ke instance sebagai volume data, menyalin data yang diperlukan ke dalamnya, dan kemudian melepaskannya. Volume harus berada di Zona Ketersediaan yang sama dengan instans.

Volume root pengganti mempertahankan ukuran, jenis, dan atribut lain yang Anda konfigurasikan saat Anda membuat dan menyiapkannya. Atribut hapus pada penghentian dari volume root asli tidak ditransfer ke volume root pengganti.

Pertimbangan untuk menggunakan volume Amazon EBS yang ada

- Volume pengganti harus berada di Zona Ketersediaan yang sama dengan instans.
- Volume pengganti tidak boleh dilampirkan ke instance apa pun pada saat permintaan.
- Volume penggantian harus di `available` negara bagian.
- Jika volume penggantian tidak memenuhi persyaratan ini, permintaan penggantian volume root gagal.

- Anda dapat menggunakan opsi ini dengan AWS Marketplace instans, tetapi kode produk pasar dan kode penagihan pada volume pengganti harus sesuai dengan kode pada instans.
- Jika volume root asli dienkripsi, volume pengganti juga harus dienkripsi. Anda dapat berpindah dari volume yang tidak dienkripsi ke yang terenkripsi, tetapi tidak dari terenkripsi ke yang tidak terenkripsi. Saat volume pengganti dienkripsi, prinsipal IAM Anda harus memiliki AWS KMS izin yang diperlukan untuk melepaskan volume asli. Anda juga harus memiliki izin untuk melampirkan dan mendekripsi volume pengganti. Ini adalah izin yang sama yang diperlukan untuk melampirkan volume terenkripsi.

Anda dapat memilih apakah akan mempertahankan volume root asli setelah proses penggantian volume root selesai. Jika Anda memilih menghapus volume root asli setelah proses penggantian selesai, volume root asli secara otomatis dihapus dan menjadi tidak dapat dipulihkan. Jika Anda memilih untuk menyimpan volume root asli setelah proses selesai, volume tetap disediakan di akun Anda; Anda harus menghapus volume secara manual ketika Anda tidak lagi membutuhkannya.

Transisi tugas penggantian volume root melalui status berikut:

- `pending`— Volume pengganti sedang dibuat.
- `in-progress`- Volume asli sedang dilepaskan dan volume pengganti sedang dilampirkan.
- `succeeded`- Volume pengganti telah berhasil dilampirkan ke instance dan instans tersedia.
- `failing`Tugas penggantian sedang dalam proses gagal.
- `failed`- Tugas penggantian gagal, tetapi volume root masih terpasang.
- `failing-detached`- Tugas penggantian sedang dalam proses gagal dan instance mungkin tidak memiliki volume root yang terpasang.
- `failed-detached`- Tugas penggantian gagal dan instance tidak memiliki volume root yang terpasang.

Jika tugas penggantian volume root gagal, instans di-boot ulang dan volume root asli tetap melekat pada instans.

## Pertimbangan-pertimbangan

Sebelum Anda mulai, pertimbangkan hal berikut.

Persyaratan

- Instans harus berada dalam status `running`.

- Instans secara otomatis di-reboot selama proses. Isi memori (RAM) dihapus saat boot ulang. Tidak diperlukan boot ulang manual.
- Anda tidak dapat mengganti volume root jika merupakan volume penyimpanan instans. Hanya instans dengan volume root Amazon EBS yang didukung.
- Anda dapat mengganti volume root untuk semua tipe instans virtual dan instans bare metal EC2. Tidak ada jenis instance bare metal lainnya yang didukung.
- Saat menggunakan snapshot, Anda hanya dapat menggunakan snapshot yang Anda buat langsung dari volume root instans saat ini atau sebelumnya.
- Jika akun Anda diaktifkan untuk enkripsi Amazon EBS secara default di Wilayah saat ini, volume root pengganti yang dibuat oleh tugas penggantian volume root selalu dienkripsi, terlepas dari status enkripsi snapshot yang ditentukan atau volume root AMI yang ditentukan.

Hasil enkripsi

Tabel berikut merangkum kemungkinan hasil enkripsi.

| | Volume root asli | Snapshot, AMI, atau volume yang ditentukan | Enkripsi secara default | Volume root pengganti | Kunci enkripsi yang digunakan untuk penggantian volume root |
|---|---|---|---|---|---|
| Mengembalikan volume root pengganti ke status peluncuran awal | Dienkripsi | Tidak berlaku | Tidak dipertimbangkan | Dienkripsi | Kunci KMS yang sama dengan volume root asli |
| | Tidak terenkripsi | Tidak berlaku | Nonaktif | Tidak terenkripsi | Tidak berlaku |
| | Tidak terenkripsi | Tidak berlaku | Aktif | Dienkripsi | Kunci KMS default akun untuk enkripsi Amazon EBS |

| | Volume root asli | Snapshot, AMI, atau volume yang ditentukan | Enkripsi secara default | Volume root pengganti | Kunci enkripsi yang digunakan untuk penggantian volume root |
|---|---|---|---|---|---|
| Mengembalikan volume root pengganti dari snapshot atau AMI | Dienkripsi | Tidak terenkripsi | Tidak dipertimbangkan | Dienkripsi | Kunci KMS yang sama dengan volume root asli |
| | Dienkripsi | Dienkripsi | Tidak dipertimbangkan | Dienkripsi | Kunci KMS yang sama dengan volume root asli |
| | Tidak terenkripsi | Tidak terenkripsi | Nonaktif | Tidak terenkripsi | Tidak berlaku |
| | Tidak terenkripsi | Tidak terenkripsi | Aktif | Dienkripsi | Kunci KMS default akun untuk enkripsi Amazon EBS |

| | Volume root asli | Snapshot, AMI, atau volume yang ditentukan | Enkripsi secara default | Volume root pengganti | Kunci enkripsi yang digunakan untuk penggantian volume root |
|---|---|---|---|---|---|
| | Tidak terenkripsi | Dienkripsi | Tidak dipertimbangkan | Dienkripsi | Jika AMI atau snapshot dimiliki oleh akun, volume pengganti dienkripsi dengan AMI atau kunci KMS snapshot. Jika AMI atau snapshot dibagikan dengan akun, volume pengganti dienkripsi dengan kunci KMS default akun untuk enkripsi Amazon EBS. |
| Kembalikan volume root pengganti dari volume Amazon EBS yang ada | Dienkripsi | Dienkripsi | Tidak berlaku | Dienkripsi | Kunci KMS dari volume yang ditentukan |

| | Volume root asli | Snapshot, AMI, atau volume yang ditentukan | Enkripsi secara default | Volume root pengganti | Kunci enkripsi yang digunakan untuk penggantian volume root |
|---|---|---|---|---|---|
| | Dienkripsi | Tidak terenkripsi | Tidak berlaku | Tidak diperbolehkan. Anda tidak dapat mengganti volume root terenkripsi dengan volume yang tidak dienkripsi. Amazon EC2 menolak tugas penggantian. | Tidak berlaku |
| | Tidak terenkripsi | Tidak terenkripsi | Tidak berlaku | Tidak terenkripsi | Tidak berlaku |
| | Tidak terenkripsi | Dienkripsi | Tidak berlaku | Dienkripsi | Kunci KMS dari volume yang ditentukan |

## Mengganti volume root

Saat Anda mengganti volume root untuk suatu instans, tugas penggantian volume root dibuat. Anda dapat menggunakan tugas penggantian volume root untuk memantau kemajuan dan hasil dari proses penggantian.

Console

Untuk mengganti volume root

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans untuk menggantikan volume root dan pilih Tindakan, Memantau dan memecahkan masalah, Ganti volume root.

   > **Note**
   >
   > Tindakan Ganti volume root dinonaktifkan jika instans yang dipilih tidak dalam status `running`.

4. Untuk Restore, pilih salah satu opsi berikut:
   - Status peluncuran — Kembalikan volume root pengganti dari snapshot yang digunakan untuk membuat volume root saat ini.
   - Snapshot — Kembalikan volume root pengganti ke snapshot yang Anda tentukan. Untuk Snapshot, pilih snapshot yang akan digunakan.
   - Gambar — Kembalikan volume root pengganti menggunakan AMI yang Anda tentukan. Untuk Gambar, pilih AMI yang akan digunakan.
5. (Opsional) Untuk tingkat inisialisasi Volume, Anda dapat menentukan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume (tingkat inisialisasi volume), di MiB/s, di mana blok snapshot akan diunduh dari Amazon S3 ke volume. Untuk informasi selengkapnya, lihat Menggunakan Tingkat Penyediaan Amazon EBS untuk Inisialisasi Volume. Untuk menggunakan tingkat inisialisasi default atau pemulihan snapshot cepat (jika diaktifkan untuk snapshot yang dipilih), jangan tentukan tingkat.
6. (Opsional) Untuk menghapus volume root yang Anda ganti, pilih H apus volume root yang diganti.
7. Pilih Buat tugas pengganti.
8. Untuk memantau tugas penggantian, pilih tab Peny impanan untuk instance dan perluas Tugas penggantian volume root terbaru.

AWS CLI

Untuk mengembalikan volume root pengganti ke status peluncuran awal

Gunakan perintah create-replace-root-volume-task. Untuk `--instance-id`, tentukan ID dari instans yang untuk menggantikan volume root. Hilangkan parameter `--snapshot-id` dan `--image-id`. Untuk menghapus volume root asli setelah diganti, sertakan `--delete-replaced-root-volume` dan tentukan`true`. Untuk menentukan tingkat inisialisasi volume di mana blok snapshot diunduh dari Amazon S3 ke volume, for`--volume-initialization-rate`, tentukan nilai antara `100` dan `300` MiB/s.

```
aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--delete-replaced-root-volume \
--volume-initialization-rate 150
```

Untuk memulihkan volume root pengganti ke snapshot tertentu

Gunakan perintah create-replace-root-volume-task. Untuk `--instance-id`, tentukan ID dari instans yang untuk menggantikan volume root. Untuk `--snapshot-id`, tentukan ID snapshot yang akan digunakan. Untuk menghapus volume root asli setelah diganti, sertakan `--delete-replaced-root-volume` dan tentukan`true`. Untuk menentukan tingkat inisialisasi volume di mana blok snapshot diunduh dari Amazon S3 ke volume, for`--volume-initialization-rate`, tentukan nilai antara `100` dan `300` MiB/s.

```
aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--snapshot-id snap-9876543210abcdef0 \
--delete-replaced-root-volume \
--volume-initialization-rate 150
```

Untuk mengembalikan volume root pengganti menggunakan AMI

Gunakan perintah create-replace-root-volume-task. Untuk `--instance-id`, tentukan ID dari instans yang untuk menggantikan volume root. Untuk `--image-id`, tentukan ID snapshot yang akan digunakan. Untuk menghapus volume root asli setelah diganti, sertakan `--delete-replaced-root-volume` dan tentukan`true`. Untuk menentukan tingkat inisialisasi volume di mana blok snapshot diunduh dari Amazon S3 ke volume, for`--volume-initialization-rate`, tentukan nilai antara `100` dan `300` MiB/s.

```
aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--image-id ami-09876543210abcdef \
--delete-replaced-root-volume \
--volume-initialization-rate 150
```

Untuk mengganti volume root menggunakan volume Amazon EBS yang ada

Gunakan perintah create-replace-root-volume-task. Untuk`--instance-id`, tentukan ID instance yang akan menggantikan volume root (misalnya,`i-1234567890abcdef0`). Untuk`--volume-id`, tentukan ID volume Amazon EBS yang akan digunakan sebagai volume root baru. Untuk menghapus volume root asli setelah penggantian selesai, sertakan`--delete-replaced-root-volume`. Amazon EC2 menerima `--volume-initialization-rate` parameter tetapi mengabaikannya dengan opsi ini karena tidak mengunduh data snapshot.

```
aws ec2 create-replace-root-volume-task \
--instance-id i-1234567890abcdef0 \
--volume-id vol-0123456789abcdef0 \
--delete-replaced-root-volume
```

Untuk melihat status tugas penggantian volume root

Gunakan perintah describe-replace-root-volume-tasks dan tentukan ID root volume pengganti tugas untuk melihat.

```
aws ec2 describe-replace-root-volume-tasks \
    --replace-root-volume-task-ids replacevol-1234567890abcdef0 \
    --query ReplaceRootVolumeTasks[].TaskState
```

Berikut ini adalah output contoh.

```
[
    "succeeded"
]
```

Atau, tentukan filter `instance-id` untuk menyaring hasil menurut instans.

```
$ aws ec2 describe-replace-root-volume-tasks \
```

```
    --filters Name=instance-id,Values=i-1234567890abcdef0
```

PowerShell

Untuk mengembalikan volume root pengganti ke status peluncuran awal

Gunakan perintah New-EC2ReplaceRootVolumeTask. Untuk `-InstanceId`, tentukan ID dari instans yang untuk menggantikan volume root. Hilangkan parameter `-SnapshotId` dan `-ImageId`. Untuk menghapus volume root asli setelah diganti, sertakan `-DeleteReplacedRootVolume` dan tentukan`$true`. Untuk menentukan tingkat inisialisasi volume di mana blok snapshot diunduh dari Amazon S3 ke volume, for`-VolumeInitializationRate`, tentukan nilai antara `100` dan `300` MiB/s.

```
New-EC2ReplaceRootVolumeTask `
    -InstanceId i-1234567890abcdef0 `
    -VolumeInitializationRate 150 `
    -DeleteReplacedRootVolume $true
```

Untuk memulihkan volume root pengganti ke snapshot tertentu

Gunakan perintah New-EC2ReplaceRootVolumeTask. Untuk `--InstanceId`, tentukan ID dari instans yang untuk menggantikan volume root. Untuk `-SnapshotId`, tentukan ID snapshot yang akan digunakan. Untuk menghapus volume root asli setelah diganti, sertakan `-DeleteReplacedRootVolume` dan tentukan`$true`. Untuk menentukan tingkat inisialisasi volume di mana blok snapshot diunduh dari Amazon S3 ke volume, for`-VolumeInitializationRate`, tentukan nilai antara `100` dan `300` MiB/s.

```
New-EC2ReplaceRootVolumeTask `
    -InstanceId i-1234567890abcdef0 `
    -SnapshotId snap-9876543210abcdef0 `
    -VolumeInitializationRate 150 `
    -DeleteReplacedRootVolume $true
```

Untuk mengembalikan volume root pengganti menggunakan AMI

Gunakan perintah New-EC2ReplaceRootVolumeTask. Untuk `-InstanceId`, tentukan ID dari instans yang untuk menggantikan volume root. Untuk `-ImageId`, tentukan ID snapshot yang akan digunakan. Untuk menghapus volume root asli setelah diganti, sertakan `-DeleteReplacedRootVolume` dan tentukan`$true`. Untuk menentukan

tingkat inisialisasi volume di mana blok snapshot diunduh dari Amazon S3 ke volume, for-`VolumeInitializationRate`, tentukan nilai antara `100` dan `300` MiB/s.

```
New-EC2ReplaceRootVolumeTask `
    -InstanceId i-1234567890abcdef0 `
    -ImageId ami-0abcdef1234567890 `
    -VolumeInitializationRate 150 `
    -DeleteReplacedRootVolume $true
```

Untuk mengganti volume root menggunakan volume Amazon EBS yang ada

Gunakan perintah New-EC2ReplaceRootVolumeTask. Untuk`-InstanceId`, tentukan ID instance yang akan menggantikan volume root (misalnya,`i-1234567890abcdef0`). Untuk`-VolumeId`, tentukan ID volume Amazon EBS yang akan digunakan sebagai volume root baru. Untuk menghapus volume root asli setelah penggantian selesai, sertakan `-DeleteReplacedRootVolume` dan tentukan`$true`. Amazon EC2 menerima `-VolumeInitializationRate` parameter tetapi mengabaikannya dengan opsi ini karena tidak mengunduh data snapshot.

```
New-EC2ReplaceRootVolumeTask `
    -InstanceId i-1234567890abcdef0 `
    -VolumeId vol-0123456789abcdef0 `
    -DeleteReplacedRootVolume $true
```

Untuk melihat status tugas penggantian volume root

Gunakan Get-EC2ReplaceRootVolumeTask perintah dan tentukan ID tugas penggantian volume root untuk dilihat.

```
(Get-EC2ReplaceRootVolumeTask `
    -ReplaceRootVolumeTaskIds replacevol-1234567890abcdef0).TaskState.Value
```

Berikut ini adalah output contoh.

```
Succeeded
```

Atau, tentukan filter `instance-id` untuk menyaring hasil menurut instans.

```
PS C:\> Get-EC2ReplaceRootVolumeTask -Filters @{Name = 'instance-id'; Values =
 'i-1234567890abcdef0'} | Format-Table
```

# Nama perangkat untuk volume di EC2 instans Amazon

Saat Anda memasang volume ke instans, Anda menyertakan nama perangkat untuk volume tersebut. Nama perangkat ini digunakan oleh Amazon EC2. Driver perangkat blok untuk instance menetapkan nama volume aktual saat memasang volume, dan nama yang ditetapkan dapat berbeda dari nama yang EC2 digunakan Amazon.

Jumlah volume yang dapat didukung oleh instans Anda mendukung ditentukan oleh sistem operasi. Untuk informasi selengkapnya, lihat Batas volume Amazon EBS untuk instans Amazon EC2.

Daftar Isi


- Nama perangkat yang tersedia
- Pertimbangan nama perangkat


## Nama perangkat yang tersedia

Instans Linux

Ada dua jenis virtualisasi yang tersedia untuk instans Linux: paravirtual (PV) dan mesin virtual perangkat keras (HVM). Jenis virtualisasi suatu instans ditentukan oleh AMI yang digunakan untuk meluncurkan instans. Semua jenis instance mendukung AMIs HVM. Beberapa jenis instance generasi sebelumnya mendukung PV AMIs. Pastikan untuk mencatat jenis virtualisasi AMI Anda karena nama perangkat yang disarankan dan tersedia yang dapat Anda gunakan bergantung pada jenis virtualisasi instans Anda. Untuk informasi selengkapnya, lihat Tipe virtualisasi.

Tabel berikut mencantumkan nama perangkat yang tersedia yang dapat Anda tentukan dalam pemetaan perangkat blok atau saat melampirkan volume EBS.

| Tipe virtualisasi | Available | Terpesan untuk volume root | Direkomendasikan untuk volume data EBS | Volume penyimpanan instans |
|---|---|---|---|---|
| Paravirtual | /dev/sd[a-z]<br>/dev/sd[a-z][1-15] | /dev/sda1 | /dev/sd[f-p]<br>/dev/sd[f-p][1-6] | /dev/sd[b-e] |

| Tipe virtualisasi | Available | Terpesan untuk volume root | Direkomendasikan untuk volume data EBS | Volume penyimpanan instans |
|---|---|---|---|---|
| | /dev/hh[a-z]<br><br>/dev/hh[a-z][1-15] | | | |
| HVM | /dev/sd[a-z]<br><br>/dev/xvd [a-c] [a-z]<br><br>/dev/xvdd [a-x] | Dibedakan berdasarkan AMI<br><br>/dev/sda1 or /dev/xvda | /dev/sd [b-z]<br><br>/dev/xvdb [b-z]<br><br>* | /dev/sd[b-e]<br><br>/dev/sd[b-h] (h1.16xlarge)<br><br>/dev/sd[b-y] (d2.8xlarge)<br><br>/dev/sd[b-i] (i2.8xlarge)<br><br>** |

* Nama perangkat yang Anda tentukan untuk volume NVMe EBS dalam pemetaan perangkat blok diganti namanya menggunakan nama NVMe perangkat (). `/dev/nvme[0-26]n1` Driver perangkat blok dapat menetapkan nama NVMe perangkat dalam urutan yang berbeda dari yang Anda tentukan untuk volume dalam pemetaan perangkat blok.

** Volume penyimpanan NVMe instance secara otomatis disebutkan dan diberi nama perangkat. NVMe

Instans Windows

AWS Windows AMIs menggunakan salah satu set driver berikut untuk mengizinkan akses ke perangkat keras virtual:

- AWS PV: Driver paravirtual untuk instans Windows
- AWS NVMe: NVMe driver

Nama perangkat untuk instance berbasis Nitro

Tabel berikut mencantumkan nama perangkat yang tersedia yang dapat Anda tentukan dalam pemetaan perangkat blok atau saat melampirkan volume EBS ke instance berbasis Nitro.

| Jenis driver | Available | Terpesan untuk volume root | Direkomendasikan untuk volume EBS | Volume penyimpanan instans |
|---|---|---|---|---|
| AWS NVMe | xvd[a-e]<br>xvd [a-c] [a-z]<br>xvdd [a-x]<br>/dev/sda1 | /dev/sda1 | xvd[b-z]<br>xvdb [b-z] | * |

* Volume penyimpanan NVMe instance secara otomatis disebutkan dan diberi huruf drive Windows.

Nama perangkat untuk instance berbasis Xen

Tabel berikut mencantumkan nama perangkat yang tersedia yang dapat Anda tentukan dalam pemetaan perangkat blok atau saat melampirkan volume EBS ke instance berbasis Xen.

| Jenis driver | Available | Terpesan untuk volume root | Direkomendasikan untuk volume EBS | Volume penyimpanan instans |
|---|---|---|---|---|
| AWS PV | xvd[b-z]<br>xvd[b-c][a-z]<br>/dev/sda1<br>/dev/sd[b-e] | /dev/sda1 | xvd [f-z] | xvdc[a-x]<br>xvd[a-e] |
| Citrix PV (tidak lagi didukung) | xvd[b-z]<br>xvd[b-c][a-z] | /dev/sda1 | xvd [f-z] | xvdc[a-x]<br>xvd[a-e] |

| Jenis driver | Available | Terpesan untuk volume root | Direkomendasikan untuk volume EBS | Volume penyimpanan instans |
|---|---|---|---|---|
| | /dev/sda1<br>/dev/sd[b-e] | | | |
| Red Hat PV (tidak lagi didukung) | xvd[a-e]<br>xvd[b-c][a-z]<br>/dev/sda1<br>/dev/sd[b-e] | /dev/sda1 | xvd[f-z] * | xvdc[a-x]<br>xvd[a-e] |

Untuk informasi selengkapnya tentang volume penyimpanan instans, lihat Penyimpanan blok sementara instans untuk instans EC2. Untuk informasi selengkapnya tentang volume NVMe EBS (instans berbasis Nitro), termasuk cara mengidentifikasi perangkat EBS, lihat Amazon EBS dan di NVMe Panduan Pengguna Amazon EBS.

## Pertimbangan nama perangkat

Ingatlah hal-hal berikut ini saat memilih nama perangkat:

- Bagian akhir dari nama perangkat yang Anda gunakan tidak boleh tumpang tindih karena dapat menyebabkan masalah saat Anda memulai instance Anda. Misalnya, hindari menggunakan kombinasi seperti `/dev/xvdf` dan `xvdf` untuk volume yang dilampirkan pada instance yang sama.
- Meskipun Anda dapat memasang volume EBS menggunakan nama perangkat yang digunakan untuk lampirkan volume penyimpanan instans, kami sangat menyarankan agar Anda tidak melakukannya karena perilaku tersebut tidak dapat diprediksi.
- Jumlah volume penyimpanan NVMe instance untuk sebuah instance tergantung pada ukuran instance. NVMe volume penyimpanan instance secara otomatis disebutkan dan diberi nama NVMe perangkat (instance Linux) atau huruf drive Windows (instance Windows).
- (Instans Windows) AWS Windows AMIs dilengkapi dengan perangkat lunak tambahan yang menyiapkan instance saat pertama kali boot. Ini adalah layanan EC2 Config (Windows AMIs sebelum Windows Server 2016) atau EC2 Launch (Windows Server 2016 dan yang lebih

baru). Setelah perangkat dipetakan ke drive, perangkat diinisialisasi dan dipasang. Drive root diinisialisasi dan dipasang sebagai `C:\`. Secara default, ketika volume EBS dipasang ke instans Windows,dapat muncul sebagai huruf pendorong pada instans tersebut. Anda dapat mengubah pengaturan untuk mengatur huruf drive volume sesuai dengan spesifikasi Anda. Misalnya volume toko, defaultnya tergantung pada driver. AWS Driver PV dan driver Citrix PV menetapkan volume penyimpanan instance huruf drive dari Z: ke A:. Driver Red Hat menetapkan volume penyimpanan instans huruf drive dari D: ke Z:. Untuk informasi selengkapnya, lihat Agen peluncuran Windows di instans Windows Amazon EC2, dan Cara volume dilampirkan dan dipetakan untuk instans Windows Amazon EC2.

- (Instance Linux) Tergantung pada driver perangkat blok kernel, perangkat dapat dilampirkan dengan nama yang berbeda dari yang Anda tentukan. Misalnya, jika Anda menentukan nama perangkat `/dev/sdh`, perangkat Anda dapat diganti namanya `/dev/xvdh` atau `/dev/hdh`. Dalam kebanyakan kasus, huruf di belakangnya tetap sama. Pada beberapa versi Red Hat Enterprise Linux (dan variannya, seperti CentOS), huruf di belakangnya dapat berubah (`/dev/sda` dapat menjadi `/dev/xvde`). Dalam hal ini, huruf yang ada dari setiap nama perangkat bertambah beberapa kali. Misalnya, jika `/dev/sdb` berganti nama `/dev/xvdf`, kemudian `/dev/sdc` berganti nama `/dev/xvdg`. Amazon Linux membuat tautan simbolis untuk nama yang Anda tetapkan ke perangkat yang diubah namanya. Sistem operasi lainnya dapat berperilaku berbeda.
- (Instance Linux) HVM AMIs tidak mendukung penggunaan nomor tambahan pada nama perangkat, kecuali untuk`/dev/sda1`, yang dicadangkan untuk volume root, dan. `/dev/sda2` Meskipun `/dev/sda2` dapat digunakan, kami tidak menyarankan penggunaan pemetaan perangkat ini dengan instans HVM.
- (Instance Linux) Saat menggunakan PV AMIs, Anda tidak dapat melampirkan volume yang berbagi huruf perangkat yang sama baik dengan maupun tanpa digit tambahan. Misalnya, jika Anda memasang volume sebagai `/dev/sdc` dan volume lainnya `/dev/sdc1`, hanya `/dev/sdc` dapat dilihat oleh instans. Untuk menggunakan digit akhir pada nama perangkat, Anda harus menggunakan digit akhir pada semua nama perangkat yang menggunakan huruf dasar yang sama (seperti `/dev/sdc1`, `/dev/sdc2`, `/dev/sdc3`).
- (Instance Linux) Beberapa kernel kustom mungkin memiliki batasan yang membatasi penggunaan untuk `/dev/sd[f-p]` atau. `/dev/sd[f-p][1-6]` Jika Anda mengalami masalah saat menggunakan `/dev/sd[q-z]` atau `/dev/sd[q-z][1-6]`, coba beralih ke `/dev/sd[f-p]` atau `/dev/sd[f-p][1-6]`.

Sebelum Anda menentukan nama perangkat yang telah Anda pilih, verifikasi bahwa itu tersedia. Jika tidak, Anda akan mendapatkan kesalahan bahwa nama perangkat sudah digunakan. Untuk

melihat perangkat disk dan titik pemasangannya, gunakan lsblk perintah (instance Linux), atau utilitas Manajemen Disk atau diskpart perintah (instance Windows).

# Memblokir pemetaan perangkat untuk volume pada instans Amazon EC2

Setiap instans yang Anda luncurkan memiliki volume root terkait, yaitu volume Amazon EBS atau volume penyimpanan instans. Anda dapat menggunakan pemetaan perangkat blok untuk menentukan volume EBS tambahan atau volume penyimpanan instans untuk dilampirkan ke instance saat diluncurkan. Anda juga dapat melampirkan volume EBS tambahan ke instance yang sedang berjalan. Namun, satu-satunya cara untuk melampirkan volume penyimpanan instans ke instance adalah dengan menggunakan pemetaan perangkat blok untuk melampirkan volume saat instance diluncurkan.

Daftar Isi


- Konsep pemetaan perangkat blok
- Menambahkan pemetaan perangkat blok ke AMI
- Tambahkan pemetaan perangkat blok ke instans Amazon EC2


## Konsep pemetaan perangkat blok

Perangkat blok adalah perangkat penyimpanan yang memindahkan data dalam urutan byte atau(blok). Perangkat ini mendukung akses acak dan umumnya menggunakan buffered I/O. Contohnya termasuk hard disk, CD-ROM drive, dan flash drive. Perangkat blok dapat dipasang secara fisik ke komputer atau diakses dari jarak jauh seolah-olah perangkat tersebut terpasang secara fisik ke komputer.

Amazon EC2 mendukung dua jenis perangkat blok:

- Volume penyimpanan instans (perangkat virtual yang perangkat keras yang mendasari secara fisik terpasang ke komputer host untuk instans)
- Volume EBS (perangkat penyimpanan jarak jauh)

pemetaan perangkat blok menentukan perangkat blok (volume penyimpanan instans dan volume EBS) untuk dilampirkan ke suatu instans. Anda dapat menentukan pemetaan perangkat blok

sebagai bagian dari pembuatan AMI sehingga pemetaan tersebut digunakan oleh semua instans yang diluncurkan dari AMI. Atau, Anda dapat menentukan pemetaan perangkat blok ketika Anda meluncurkan instans, sehingga pemetaan ini menimpa pemetaan yang ditentukan dalam AMI tempat Anda meluncurkan instans. Perhatikan bahwa semua volume penyimpanan instans NVMe nvolume yang didukung oleh tipe instans secara otomatis dienumerasi dan diberi nama perangkat pada peluncuran instans; memasukkannya dalam pemetaan perangkat blok Anda tidak berpengaruh.

Daftar Isi


- Entri pemetaan perangkat blok
- Peringatan penyimpanan instans pemetaan perangkat pemetaan perangkat blok
- Contoh pemetaan perangkat blok
- Cara perangkat disediakan dalam sistem operasi


## Entri pemetaan perangkat blok

Ketika Anda membuat pemetaan perangkat blok, Anda menentukan informasi berikut untuk setiap perangkat blok yang perlu dilampirkan ke instans:

- Nama perangkat yang digunakan Amazon EC2. Driver perangkat blok untuk instans menetapkan nama volume aktual saat melakukan pemasangan volume. Nama yang diberikan dapat berbeda dari nama yang direkomendasikan oleh Amazon EC2. Untuk informasi selengkapnya, lihat Nama perangkat untuk volume di EC2 instans Amazon.

Untuk volume penyimpanan instans, Anda juga menentukan informasi berikut:

- Perangkat virtual: `ephemeral[0-23]`. Perhatikan bahwa jumlah dan ukuran volume penyimpanan instans yang tersedia untuk instans Anda berbeda-beda menurut tipe instans.

Untuk instans volume penyimpanan instans NVMe, informasi berikut juga berlaku:

- Volume ini secara otomatis dienumerasi dan diberi nama perangkat; menyertakannya dalam pemetaan perangkat blok Anda tidak akan berpengaruh.

Untuk volume EBS, Anda juga menentukan informasi berikut:

- ID snapshot yang digunakan untuk membuat perangkat blok (snap-xxxxxxxx). Nilai ini opsional selama Anda menentukan ukuran volume. Anda tidak dapat menentukan ID snapshot yang diarsipkan.
- Ukuran volume, dalam GiB. Ukuran yang ditentukan harus lebih besar atau sama dengan ukuran snapshot yang ditentukan.
- Apakah akan menghapus volume pada saat pengakhiran instans (`true` atau `false`). Nilai default adalah `true` untuk volume root dan `false` untuk volume terlampir. Saat Anda membuat AMI, sistem pemetaan perangkat blok mewarisi pengaturan ini dari instans. Saat diluncurkan, instans akan mewarisi pengaturan ini dari AMI.
- Tipe volume, yang bisa berupa `gp2` dan `gp3` untuk SSD Tujuan Umum, `io1` dan `io2` untuk SSD IOPS yang Tersedia, `st1` untuk HDD Throughput Dioptimalkan, `sc1` untuk Cold HDD, atau `standard` untuk Magnetik.
- Jumlah input/output operasi per detik (IOPS) yang didukung volume. (Hanya digunakan dengan volume `io1` dan `io2`.)
- Beberapa jenis instans mendukung lebih dari satu kartu EBS. Anda dapat memilih kartu EBS untuk volume yang akan dilampirkan dengan menentukan indeks kartu EBS. Untuk informasi selengkapnya, lihat Kartu EBS.

## Peringatan penyimpanan instans pemetaan perangkat pemetaan perangkat blok

Ada beberapa peringatan yang perlu dipertimbangkan saat meluncurkan instans dengan AMI yang memiliki penyimpanan instans dalam pemetaan perangkat blok.

- Beberapa tipe instans menyertakan lebih banyak volume penyimpanan instans daripada yang lain, dan beberapa tipe instans tidak mengandung volume penyimpanan instans sama sekali. Jika tipe instans Anda mendukung satu volume penyimpanan instans, dan AMI Anda memiliki pemetaan untuk dua volume penyimpanan instans, instans akan meluncurkan dengan satu volume penyimpanan instans.
- Volume penyimpanan instans hanya dapat dipetakan pada waktu peluncuran. Anda tidak dapat menghentikan instans tanpa volume penyimpanan instans (seperti `t2.micro`), mengubah instans ke tipe yang mendukung volume penyimpanan instans, lalu memulai ulang instans dengan volume penyimpanan instans. Namun, Anda dapat membuat AMI dari instans dan meluncurkannya pada tipe instans yang mendukung volume penyimpanan instans, dan memetakan volume penyimpanan instans tersebut ke instans.

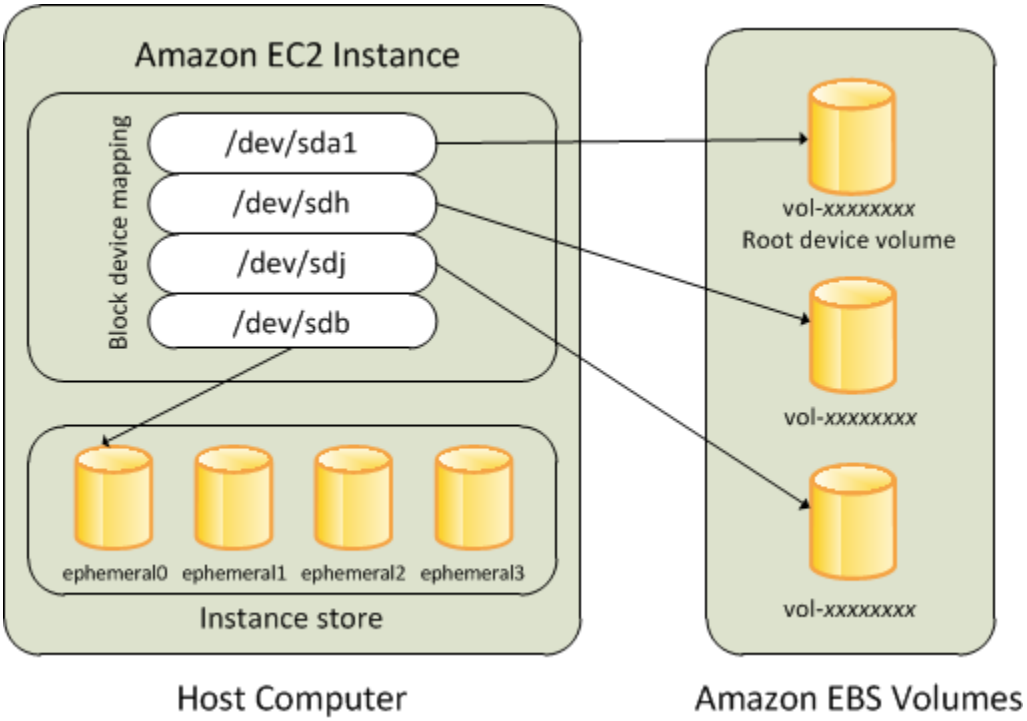
- Jika Anda meluncurkan instans dengan volume penyimpanan instans yang dipetakan, lalu menghentikan instans dan mengubahnya menjadi tipe instans dengan volume penyimpanan instans yang lebih sedikit, lalu memulai ulang instans tersebut, pemetaan volume penyimpanan instans dari peluncuran awal akan tetap muncul di metadata instans. Namun, hanya jumlah maksimum volume penyimpanan instans yang didukung untuk tipe instans tersebut yang tersedia untuk instans tersebut.

  > **Note**
  >
  > Saat instans dihentikan atau diakhiri, semua data pada volume penyimpanan instans akan hilang.

- Bergantung pada kapasitas penyimpanan instans pada saat peluncuran, instans M3 dapat mengabaikan pemetaan perangkat blok penyimpanan instans AMI pada saat peluncuran kecuali jika ditentukan pada saat peluncuran. Anda harus menentukan instans pemetaan perangkat blok penyimpanan pada saat peluncuran, bahkan jika AMI yang Anda luncurkan memiliki volume penyimpanan yang dipetakan di AMI, untuk memastikan bahwa volume penyimpanan instans tersedia saat peluncuran.

## Contoh pemetaan perangkat blok

Gambar ini menunjukkan contoh pemetaan perangkat blok untuk sebuah EBS-backed instance. Gambar ini memetakan `/dev/sdb` ke `ephemeral0` dan memetakan dua volume EBS, satu untuk `/dev/sdh` dan yang lainnya ke `/dev/sdj`. Ini juga menunjukkan volume EBS yang merupakan volume akar,`/dev/sda1`.




Perhatikan bahwa contoh pemetaan perangkat blok ini digunakan dalam contoh perintah dan API dalam topik ini. Anda dapat menemukan contoh perintah dan API yang membuat pemetaan perangkat blok di Tentukan pemetaan perangkat blok untuk AMI dan Memperbarui pemetaan perangkat blok saat meluncurkan suatu instans.

## Cara perangkat disediakan dalam sistem operasi

Nama perangkat seperti `/dev/sdh` dan `xvdh` digunakan oleh Amazon EC2 untuk menjelaskan perangkat blok. Pemetaan perangkat blok digunakan oleh Amazon EC2 untuk menentukan perangkat blok yang akan dilampirkan di instans EC2. Setelah dilampirkan pada suatu instans, perangkat blok harus dipasang oleh sistem operasi sebelum Anda dapat mengakses perangkat penyimpanan. Ketika dilepaskan dari suatu instans, perangkat blok dilepaskan oleh sistem operasi dan Anda tidak dapat lagi mengakses perangkat penyimpanan.

Instans Linux — Nama perangkat yang ditentukan dalam pemetaan perangkat blok dipetakan ke perangkat blok yang sesuai saat instance pertama kali di-boot. Tipe instans menentukan volume penyimpanan instans mana yang diformat dan dipasang secara default. Anda dapat memasang volume penyimpanan instans tambahan saat peluncuran, selama Anda tidak melebihi jumlah volume penyimpanan instans yang tersedia untuk tipe instans Anda. Untuk informasi selengkapnya, lihat Penyimpanan blok sementara instans untuk instans EC2. Driver perangkat blok untuk instans menentukan perangkat yang digunakan saat volume diformat dan dipasang.

Instans Windows — Nama perangkat yang ditentukan dalam pemetaan perangkat blok dipetakan ke perangkat blok yang sesuai ketika instance pertama kali di-boot, dan kemudian layanan Ec2Config menginisialisasi dan memasang drive. Volume root dipasang sebagai`C:\`. Volume penyimpanan instans dipasang sebagai `Z:\`, `Y:\`, dan sebagainya. Saat dipasang, volume EBS dapat dipasang menggunakan huruf drive yang tersedia. Namun, Anda dapat mengonfigurasi bagaimana huruf drive ditetapkan ke volume EBS; untuk informasi selengkapnya, lihatthe section called “Agen peluncuran Windows”.

## Menambahkan pemetaan perangkat blok ke AMI

Setiap AMI memiliki pemetaan perangkat blok yang menentukan perangkat blok yang akan dipasang ke suatu instans ketika diluncurkan dari AMI. Untuk menambahkan lebih banyak perangkat blok ke AMI, Anda harus membuat AMI sendiri.

Daftar Isi


- Tentukan pemetaan perangkat blok untuk AMI
- Lihat volume EBS dalam pemetaan perangkat blok AMI


### Tentukan pemetaan perangkat blok untuk AMI

Ada dua cara untuk menentukan volume sebagai tambahan pada volume root saat Anda membuat AMI. Jika Anda telah melampirkan volume ke instans yang sedang berjalan sebelum membuat AMI dari instans, pemetaan perangkat blok untuk AMI akan menyertakan volume yang sama. Untuk volume EBS, data yang ada disimpan ke snapshot baru, dan snapshot baru inilah yang ditentukan dalam pemetaan perangkat blok. Untuk volume penyimpanan instans, data tidak disimpan.

Untuk EBS-backed AMI, Anda dapat menambahkan volume EBS dan volume penyimpanan instans menggunakan pemetaan perangkat blok. Untuk Amazon S3-backed AMI, Anda dapat menambahkan volume penyimpanan instans hanya dengan memodifikasi entri pemetaan perangkat blok dalam file manifes gambar saat mendaftarkan gambar.

> **Note**
>
> Untuk instans M3, Anda harus menentukan volume penyimpanan instans dalam pemetaan perangkat blok untuk instans ketika Anda meluncurkannya. Saat Anda meluncurkan instans M3, volume penyimpanan instans yang ditentukan dalam pemetaan perangkat blok untuk

AMI dapat diabaikan jika tidak ditentukan sebagai bagian dari pemetaan perangkat blok instans.

### Console

Untuk menambahkan volume ke AMI

1. Buka konsol Amazon EC2.
2. Di panel navigasi, pilih Instans.
3. Pilih suatu instans dan pilih Tindakan, Citra dan templat, Buat citra.
4. Masukkan nama dan deskripsi untuk citra.
5. Volume instans muncul di bawah Volume instans. Untuk menambahkan volume lainnya, pilih Tambahkan volume.
6. Untuk Tipe volume, pilih tipe volume. Untuk Perangkat pilih nama perangkat. Untuk volume EBS, Anda dapat menentukan detail tambahan, seperti snapshot, ukuran volume, tipe volume, IOPS, dan status enkripsi.
7. Pilih Buat citra.

### AWS CLI

Untuk menambahkan volume ke AMI

Gunakan perintah create-image untuk menentukan pemetaan perangkat blok untuk AMI. EBS-backed Gunakan perintah register-image untuk menentukan pemetaan perangkat blok untuk Amazon S3-backed AMI.

Tentukan pemetaan perangkat blok menggunakan parameter `--block-device-mappings`. Anda dapat menentukan argumen yang dikodekan dalam JSON langsung pada baris perintah atau dengan mereferensikan file JSON, seperti yang ditunjukkan di sini.

```
--block-device-mappings file://mapping.json
```

Untuk menambahkan volume penyimpanan instans, gunakan pemetaan berikut. Perhatikan bahwa volume penyimpanan instans NVMe ditambahkan secara otomatis.

```
{
    "DeviceName": "device_name",
```

```
    "VirtualName": "ephemeral0"
}
```

Untuk menambahkan volume kosong 100 GiB, gunakan pemetaan berikut ini.

```
{
    "DeviceName": "device_name",
    "Ebs": {
      "VolumeSize": 100
    }
}
```

Untuk menambahkan volume EBS berdasarkan snapshot, gunakan pemetaan berikut.

```
{
    "DeviceName": "device_name",
    "Ebs": {
      "SnapshotId": "snap-1234567890abcdef0"
    }
}
```

Untuk pemetaan perangkat, gunakan pemetaan berikut.

```
{
    "DeviceName": "device_name",
    "NoDevice": ""
}
```

PowerShell

Gunakan New-EC2Image cmdlet untuk menentukan pemetaan perangkat blok untuk AMI. EBS-backed Gunakan Register-EC2Image cmdlet untuk menentukan pemetaan perangkat blok untuk Amazon S3-backed AMI.

Tambahkan `-BlockDeviceMapping` opsi, tentukan pembaruan dibdm:

```
-BlockDeviceMapping $bdm
```

Pemetaan berikut menambahkan volume berdasarkan snapshot.

```
$ebd = New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice
$ebd.SnapshotId = "snap-1234567890abcdef0"
```

```
$bdm = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$bdm.DeviceName = "device_name"
$bdm.Ebs = $ebd
```

Pemetaan berikut menambahkan volume kosong 100 GB.

```
$ebd = New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice
$ebd.VolumeSize = 100
$bdm = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$bdm.DeviceName = "device_name"
$bdm.Ebs = $ebd
```

## Lihat volume EBS dalam pemetaan perangkat blok AMI

Anda dapat dengan mudah mengenumerasi volume EBS dalam pemetaan perangkat blok untuk AMI.

### Console

Untuk melihat volume EBS untuk AMI

1. Buka konsol Amazon EC2.
2. Di panel navigasi, pilih AMI.
3. Pilih gambar EBS dari daftar Filter untuk mendapatkan daftar EBS-backed AMI.
4. Pilih AMI yang diinginkan, dan lihat tab Detail. Minimal, informasi berikut tersedia untuk volume root (di mana istilah perangkat root setara dengan volume root):
   - Jenis Perangkat Root (`ebs`)
   - Nama Perangkat Root (misalnya, `/dev/sda1`)
   - Perangkat blok (misalnya, `/dev/sda1=snap-1234567890abcdef0:8:true`)

   Jika AMI dibuat dengan volume EBS tambahan menggunakan pemetaan perangkat blok, Perangkat Blok menampilkan pemetaan untuk volume tambahan juga. Jika AMI dibuat dengan volume EBS tambahan menggunakan pemetaan perangkat blok, Perangkat Blok menampilkan pemetaan untuk volume tambahan juga.

### AWS CLI

Untuk melihat volume EBS untuk AMI

Gunakan perintah describe-images.

```
aws ec2 describe-images \
    --image-ids ami-0abcdef1234567890 \
    --query Image[0].BlockDeviceMappings
```

PowerShell

Untuk melihat volume EBS untuk AMI

Gunakan Get-EC2Image cmdlet.

```
(Get-EC2Image -ImageId ami-0abcdef1234567890).BlockDeviceMappings
```

## Tambahkan pemetaan perangkat blok ke instans Amazon EC2

Secara default, instans yang Anda luncurkan menyertakan perangkat penyimpanan apa pun yang ditentukan dalam pemetaan perangkat blok AMI tempat Anda meluncurkan instans. Anda dapat menentukan perubahan pada pemetaan perangkat blok untuk sebuah instans saat Anda meluncurkannya, dan pembaruan ini akan menimpa atau bergabung dengan pemetaan perangkat blok AMI.

Batasan

- Untuk volume root, Anda hanya dapat memodifikasi hal berikut: ukuran volume, tipe volume, dan tanda Hapus saat Pengakhiran.
- Saat Anda memodifikasi volume EBS, Anda tidak dapat mengurangi ukuran. Oleh karena itu, Anda harus menentukan snapshot yang ukurannya sama atau lebih besar dari ukuran snapshot yang ditentukan dalam pemetaan perangkat blok AMI.

Tugas

- Memperbarui pemetaan perangkat blok saat meluncurkan suatu instans
- Perbarui pemetaan perangkat blok instans yang berjalan
- Lihat volume EBS dalam pemetaan perangkat blok instans
- Lihat pemetaan perangkat blok instans untuk volume penyimpanan instans

## Memperbarui pemetaan perangkat blok saat meluncurkan suatu instans

Anda dapat menambahkan volume EBS dan volume penyimpanan instans ke instans pada saat Anda meluncurkannya. Perhatikan bahwa pembaruan pemetaan perangkat blok untuk suatu instans tidak membuat perubahan permanen pada pemetaan perangkat blok AMI tempatnya diluncurkan.

### Console

Untuk memperbarui volume untuk instance saat peluncuran

1. Ikuti prosedur untuk meluncurkan instance, tetapi jangan luncurkan instans sampai Anda menyelesaikan langkah-langkah berikut untuk memperbarui volume.
2. (Opsional) Untuk menambahkan volume, pilih Konfigurasi penyimpanan, Tambahkan volume baru. Pilih ukuran volume dan jenis volume.
3. (Opsional) Untuk menekan volume yang ditentukan oleh pemetaan perangkat blok AMI, pilih Konfigurasikan penyimpanan, H apus.
4. (Opsional) Untuk mengubah konfigurasi volume EBS, pada panel Konfigurasi penyimpanan, pilih L anjutan. Perluas informasi untuk volume, dan buat perubahan apa pun yang Anda butuhkan.
5. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

### AWS CLI

Untuk memperbarui volume untuk instance saat peluncuran

Gunakan perintah run-instance dengan opsi. `--block-device-mappings`

```
--block-device-mappings file://mapping.json
```

Misalnya, pemetaan perangkat blok AMI menentukan hal berikut:

- `/dev/xvda`- Volume akar EBS
- `/dev/sdh`- Volume EBS dibuat dari `snap-1234567890abcdef0`
- `/dev/sdj`- Volume EBS kosong dengan ukuran `100`
- `/dev/sdb`- Volume penyimpanan instans `ephemeral0`

Misalkan berikut ini adalah pemetaan perangkat blok instance di`mapping.json`.

```
[
    {
        "DeviceName": "/dev/xvda",
        "Ebs": {
            "VolumeSize": 100
        }
    },
    {
        "DeviceName": "/dev/sdj",
        "NoDevice": ""
    },
    {
        "DeviceName": "/dev/sdh",
        "Ebs": {
            "VolumeSize": 300
        }
    },
    {
        "DeviceName": "/dev/sdc",
        "VirtualName": "ephemeral1"
    }
]
```

Pemetaan perangkat blok instance melakukan hal berikut:

- Mengganti ukuran volume root,`/dev/xvda`, meningkatkannya menjadi 100 GiB.
- M `/dev/sdj` encegah melampirkan ke instance.
- Mengganti ukuran`/dev/sdh`, meningkatkannya menjadi 300 GiB. Perhatikan bahwa Anda tidak perlu menentukan ID snapshot lagi.
- Menambahkan volume fana,. `/dev/sdc` Jika jenis instans tidak mendukung beberapa volume penyimpanan instans, ini tidak berpengaruh. Jika jenis instans mendukung volume penyimpanan instans NVMe, volume tersebut akan dihitung secara otomatis dan disertakan dalam pemetaan perangkat blok instans dan tidak dapat diganti.

PowerShell

Untuk memperbarui volume untuk instance saat peluncuran

Gunakan `-BlockDeviceMapping` parameter dengan New-EC2Instance cmdlet dengan parameter. `-BlockDeviceMapping`

```
-BlockDeviceMapping $bdm
```

Misalkan berikut ini adalah pemetaan perangkat blok instance di`$bdm`.

```
$bdm = @()

$root = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$root.DeviceName = "/dev/xvda"
$ebs1 = New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice
$ebs1.VolumeSize = 100
$root.Ebs = $ebs1
$bdm += $root

$sdj = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$sdj.DeviceName = "/dev/sdj"
$sdj.NoDevice = ""
$bdm += $sdj

$sdh = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$sdh.DeviceName = "/dev/sdh"
$ebs2 = New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice
$ebs2.VolumeSize = 300
$sdh.Ebs = $ebs2
$bdm += $sdh

$sdc = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$sdc.DeviceName = "/dev/sdc"
$sdc.VirtualName = "ephemeral1"
$bdm += $sdc
```

Pemetaan perangkat blok instance melakukan hal berikut:

- Mengganti ukuran volume root,`/dev/xvda`, meningkatkannya menjadi 100 GiB.
- M `/dev/sdj` encegah melampirkan ke instance.
- Mengganti ukuran`/dev/sdh`, meningkatkannya menjadi 300 GiB. Perhatikan bahwa Anda tidak perlu menentukan ID snapshot lagi.
- Menambahkan volume fana,. `/dev/sdc` Jika jenis instans tidak mendukung beberapa volume penyimpanan instans, ini tidak berpengaruh. Jika jenis instans mendukung volume

penyimpanan instans NVMe, volume tersebut akan dihitung secara otomatis dan disertakan dalam pemetaan perangkat blok instans dan tidak dapat diganti.

## Perbarui pemetaan perangkat blok instans yang berjalan

Anda tidak perlu menghentikan instans sebelum mengubah atribut ini.

AWS CLI

Untuk memperbarui pemetaan perangkat blok dari instance yang sedang berjalan

Gunakan perintah modify- instance-atribut.

Tambahkan `--block-device-mappings` opsi:

```
--block-device-mappings file://mapping.json
```

Di`mapping.json`, tentukan pembaruan. Misalnya, pembaruan berikut mengubah volume root untuk bertahan.

```
[
  {
    "DeviceName": "/dev/sda1",
    "Ebs": {
      "DeleteOnTermination": false
    }
  }
]
```

PowerShell

Untuk memperbarui pemetaan perangkat blok dari instance yang sedang berjalan

Gunakan Edit-EC2InstanceAttribute cmdlet.

Tambahkan `-BlockDeviceMapping` opsi:

```
-BlockDeviceMapping $bdm
```

Di`bdm`, tentukan pembaruan. Misalnya, pembaruan berikut mengubah volume root untuk bertahan.

```
$ebd = New-Object -TypeName Amazon.EC2.Model.EbsBlockDevice
$ebd.DeleteOnTermination = false
$bdm = New-Object -TypeName Amazon.EC2.Model.BlockDeviceMapping
$bdm.DeviceName = "/dev/sda1"
$bdm.Ebs = $ebd
```

## Lihat volume EBS dalam pemetaan perangkat blok instans

Anda dapat dengan mudah mengenumerasi volume EBS yang dipetakan sebagai instans.

Console

Untuk melihat volume EBS untuk sebuah instance

1. Buka konsol Amazon EC2.
2. Di panel navigasi, pilih Instans.
3. Pilih instance dan lihat detail yang ditampilkan di tab Penyimpanan. Minimal, informasi berikut tersedia untuk volume root (di mana istilah perangkat root setara dengan volume root):

   - Jenis perangkat root (misalnya, EBS)
   - Nama perangkat root (misalnya, `/dev/xvda`)
   - Perangkat blok (misalnya, `/dev/xvda`, `/dev/sdf`, dan `/dev/sdj`)

   Jika instans diluncurkan dengan volume EBS tambahan menggunakan pemetaan perangkat blok, maka akan muncul di bawah Perangkat blok. Volume penyimpanan instans tidak muncul di tab ini.
4. Untuk menampilkan informasi tambahan tentang volume EBS, pilih ID volumenya untuk membuka halaman volume.

AWS CLI

Untuk melihat volume EBS untuk sebuah instance

Gunakan perintah describe-instances.

```
aws ec2 describe-instances \
```

```
    --instance-ids i-1234567890abcdef0 \
    --query Reservations[*].Instances[0].BlockDeviceMappings
```

PowerShell

Untuk melihat volume EBS untuk sebuah instance

Gunakan Get-EC2Instance cmdlet.

```
(Get-EC2Instance -InstanceId i-0bac57d7472c89bac).Instances.BlockDeviceMappings
```

## Lihat pemetaan perangkat blok instans untuk volume penyimpanan instans

Jenis instance menentukan jumlah dan jenis volume penyimpanan instans yang tersedia untuk instance. Jika jumlah volume penyimpanan instans dalam pemetaan perangkat blok melebihi jumlah volume penyimpanan instans yang tersedia untuk sebuah instans, volume tambahan akan diabaikan. Untuk melihat volume penyimpanan instans untuk instance Anda, jalankan lsblk perintah (instans Linux) atau buka Manajemen Disk Windows (instans Windows). Untuk mempelajari berapa banyak volume penyimpanan instans yang didukung oleh setiap jenis instans, lihat Spesifikasi jenis instans Amazon EC2.

Saat Anda melihat pemetaan perangkat blok untuk instans Anda, Anda hanya dapat melihat volume EBS, bukan volume penyimpanan instans. Metode yang Anda gunakan untuk melihat volume penyimpanan instans untuk instans tergantung pada tipe volume.

Volume penyimpanan instans NVMe

Instans Linux

Anda dapat menggunakan paket baris perintah NVMe,nvme-cli, untuk mengueri volume penyimpanan instans NVMe dalam pemetaan perangkat blok. Unduh dan instal paket di instans Anda, lalu jalankan perintah berikut.

```
[ec2-user ~]$ sudo nvme list
```

Berikut ini contoh output untuk suatu instans. Teks di kolom Model menunjukkan apakah volume adalah volume EBS atau volume penyimpanan instans. Dalam contoh ini, `/dev/nvme1n1` dan `/dev/nvme2n1` adalah volume penyimpanan instans.

```
Node             SN                   Model
 Namespace
---------------- -------------------- ----------------------------------------
 ---------
/dev/nvme0n1     vol06afc3f8715b7a597 Amazon Elastic Block Store              1

/dev/nvme1n1     AWS2C1436F5159EB6614 Amazon EC2 NVMe Instance Storage        1

/dev/nvme2n1     AWSB1F4FF0C0A6C281EA Amazon EC2 NVMe Instance Storage        1
  ...
```

Instans Windows

Anda dapat menggunakan Manajemen Disk atau PowerShell untuk membuat daftar volume NVMe penyimpanan EBS dan instans. Untuk informasi selengkapnya, lihat the section called “Petakan disk NVME ke volume”.

Volume penyimpanan instans HDD atau SSD

Anda dapat menggunakan metadata instans untuk mengueri volume penyimpanan instans HDD atau SSD dalam pemetaan perangkat blok. Volume penyimpanan instans NVMe tidak disertakan.

URI dasar untuk semua permintaan untuk metadata instans adalah `http://169.254.169.254/latest/`. Untuk informasi selengkapnya, lihat Menggunakan metadata instans untuk mengelola instans EC2 Anda.

Instans Linux

Pertama, hubungkan ke instans berjalan Anda. Dari instans, gunakan kueri ini untuk mendapatkan pemetaan perangkat blok.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-
aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-
data/block-device-mapping/
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/block-device-mapping/
```

Responsnya mencakup nama-nama perangkat blok untuk instans tersebut. Misalnya, output untuk instans yang didukung oleh instans `m1.small` yang didukung penyimpanan instans terlihat seperti ini.

```
ami
ephemeral0
root
swap
```

`ami`Perangkat adalah volume root seperti yang terlihat oleh instance. Volume penyimpanan instans diberi nama `ephemeral[0-23]`. Parameter perangkat `swap` adalah untuk file halaman. Jika Anda juga telah memetakan volume EBS, volume tersebut muncul sebagai `ebs1`, `ebs2`, dan seterusnya.

Untuk mendapatkan detail tentang perangkat blok individu dalam pemetaan perangkat blok, tambahkan namanya ke kueri sebelumnya, seperti yang ditunjukkan di sini.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

Instans Windows

Pertama, hubungkan ke instans berjalan Anda. Dari instans, gunakan kueri ini untuk mendapatkan pemetaan perangkat blok.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-mapping/
```

Responsnya mencakup nama-nama perangkat blok untuk instans tersebut. Misalnya, output untuk instans yang didukung oleh instans `m1.small` yang didukung penyimpanan instans terlihat seperti ini.

```
ami
ephemeral0
root
swap
```

`ami`Perangkat adalah volume root seperti yang terlihat oleh instance. Volume penyimpanan instans diberi nama `ephemeral[0-23]`. Parameter perangkat `swap` adalah untuk file halaman. Jika Anda juga telah memetakan volume EBS, volume tersebut muncul sebagai `ebs1`, `ebs2`, dan seterusnya.

Untuk mendapatkan detail tentang perangkat blok individu dalam pemetaan perangkat blok, tambahkan namanya ke kueri sebelumnya, seperti yang ditunjukkan di sini.

```
PS C:\> Invoke-RestMethod -uri http://169.254.169.254/latest/meta-data/block-device-mapping/ephemeral0
```

# Cara volume dilampirkan dan dipetakan untuk instans Windows Amazon EC2

> **Note**
>
> Topik ini hanya berlaku untuk instance Windows.

Instans Windows Anda dilengkapi dengan volume EBS yang berfungsi sebagai volume root. Jika instance Windows Anda menggunakan AWS driver PV atau Citrix PV, Anda dapat menambahkan hingga 25 volume secara opsional, menghasilkan total 26 volume. Untuk informasi selengkapnya, lihat Batas volume Amazon EBS untuk instans Amazon EC2.

Bergantung pada tipe instans, Anda akan memiliki dari 0 hingga 24 kemungkinan volume penyimpanan instans yang tersedia untuk instans. Untuk menggunakan volume penyimpanan instans yang tersedia untuk instans, Anda harus menentukannya saat Anda membuat AMI atau meluncurkan instans. Anda juga dapat menambahkan volume EBS saat Anda membuat AMI atau meluncurkan instans, atau melampirkannya saat instans berjalan.

Saat Anda menambahkan volume ke instans, Anda menentukan nama perangkat yang digunakan Amazon EC2. Untuk informasi lebih lanjut, lihatNama perangkat untuk volume di EC2 instans Amazon. AWS Windows Amazon Machine Image (AMI) berisi sekumpulan driver yang digunakan oleh Amazon EC2 untuk memetakan instans store dan volume EBS ke disk Windows dan huruf drive.

Metode untuk memetakan disk ke volume EBS

- Petakan disk NVMe pada instans Windows Amazon EC2 ke volume
- Memetakan disk non-NVMe pada instans Windows Amazon EC2 ke volume

# Petakan disk NVMe pada instans Windows Amazon EC2 ke volume

Dengan Nitro-based instance, volume EBS diekspos sebagai perangkat NVMe. Topik ini menjelaskan cara melihat disk NVMe yang tersedia untuk sistem operasi Windows pada instans Anda. Ini juga menunjukkan cara memetakan disk NVMe tersebut ke volume Amazon EBS yang mendasarinya dan nama perangkat yang ditentukan untuk pemetaan perangkat blok yang digunakan oleh Amazon EC2.

Topik

- Daftar disk NVMe
- Petakan disk NVMe ke volume

## Daftar disk NVMe

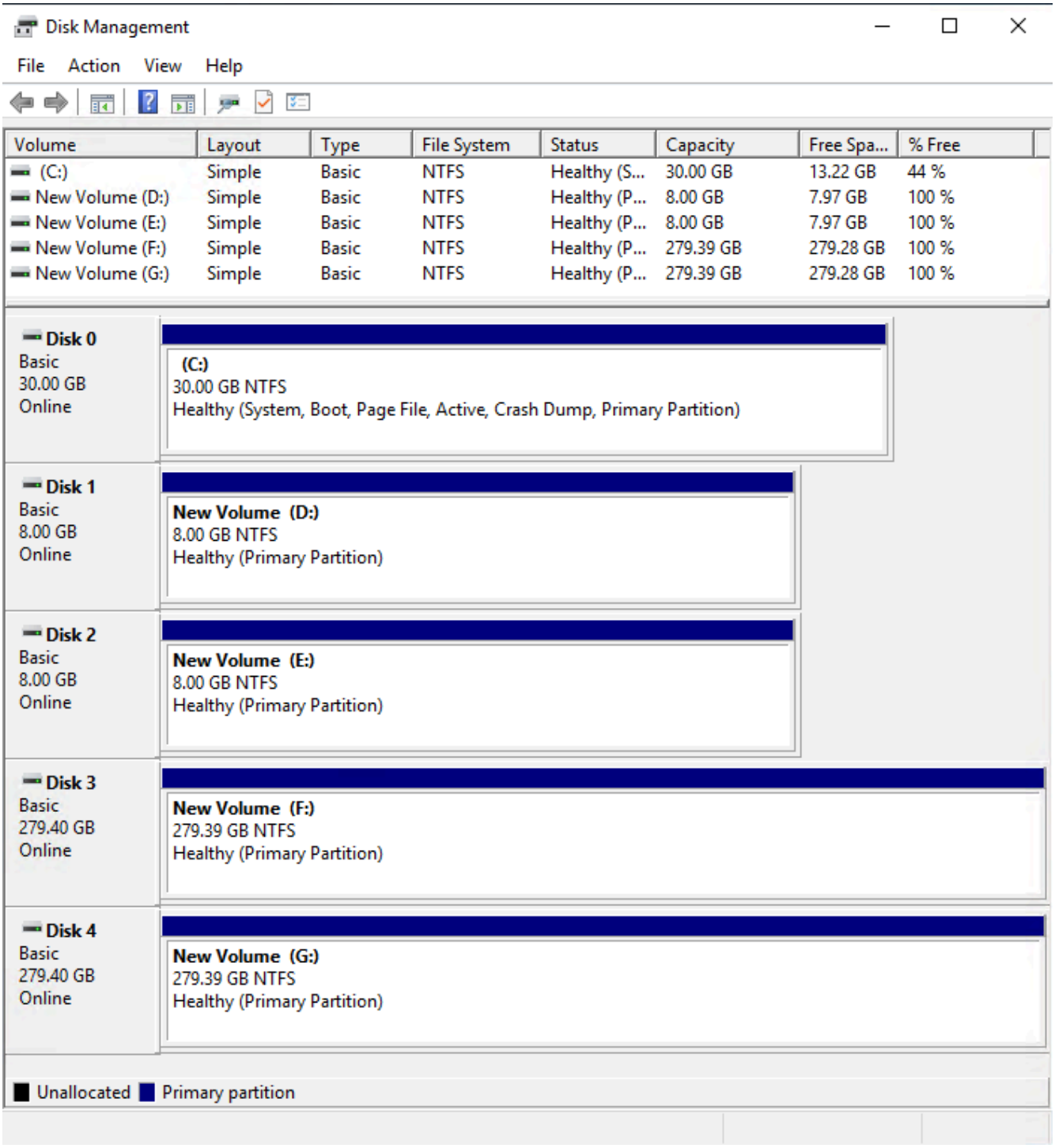
Anda dapat menemukan disk di instans Windows menggunakan Manajemen Disk atau Powershell.

Disk Management

Untuk menemukan disk di instans Windows Anda

1. Masuk ke instans Windows menggunakan Remote Desktop. Untuk informasi selengkapnya, lihat Hubungkan ke instans Windows Anda menggunakan RDP.
2. Mulai utilitas Manajemen Disk.
3. Tinjau disk. Volume root adalah volume EBS yang dipasang sebagai `C:\`. Jika tidak ada disk lain yang ditampilkan, berarti Anda tidak menentukan volume tambahan saat membuat AMI atau meluncurkan instans.

   Berikut ini adalah contoh yang menunjukkan disk yang tersedia jika Anda meluncurkan instans `r5d.4xlarge` dengan dua volume EBS tambahan.




PowerShell

PowerShell Skrip berikut mencantumkan setiap disk dan nama dan volume perangkat yang sesuai. Ini dimaksudkan untuk digunakan dengan Nitro-based instance, yang menggunakan NVMe EBS dan volume penyimpanan instans.

Hubungkan ke instans Windows Anda dan jalankan perintah berikut untuk mengaktifkan PowerShell pelaksanaan skrip.

```
Set-ExecutionPolicy RemoteSigned
```

Salin skrip berikut dan simpan sebagai `mapping.ps1` di instans Windows Anda.

```
# List the disks for NVMe volumes

function Get-EC2InstanceMetadata {
    param([string]$Path)
    (Invoke-WebRequest -Uri "http://169.254.169.254/latest/$Path").Content
}

function GetEBSVolumeId {
    param($Path)
    $SerialNumber = (Get-Disk -Path $Path).SerialNumber
    if($SerialNumber -clike 'vol*'){
        $EbsVolumeId = $SerialNumber.Substring(0,20).Replace("vol","vol-")
    }
    else {
       $EbsVolumeId = $SerialNumber.Substring(0,20).Replace("AWS","AWS-")
    }
    return $EbsVolumeId
}

function GetDeviceName{
    param($EbsVolumeId)
    if($EbsVolumeId -clike 'vol*'){

        $Device  = ((Get-EC2Volume -VolumeId $EbsVolumeId ).Attachment).Device
        $VolumeName = ""
    }
     else {
        $Device = "Ephemeral"
        $VolumeName = "Temporary Storage"
    }
    Return $Device,$VolumeName
}

function GetDriveLetter{
    param($Path)
    $DiskNumber =  (Get-Disk -Path $Path).Number
```

```
    if($DiskNumber -eq 0){
        $VirtualDevice = "root"
        $DriveLetter = "C"
        $PartitionNumber = (Get-Partition -DriveLetter C).PartitionNumber
    }
    else
    {
        $VirtualDevice = "N/A"
        $DriveLetter = (Get-Partition -DiskNumber $DiskNumber).DriveLetter
        if(!$DriveLetter)
        {
            $DriveLetter = ((Get-Partition -DiskId $Path).AccessPaths).Split(",")[0]
        }
        $PartitionNumber = (Get-Partition -DiskId $Path).PartitionNumber
    }

    return $DriveLetter,$VirtualDevice,$PartitionNumber

}

$Report = @()
foreach($Path in (Get-Disk).Path)
{
    $Disk_ID = ( Get-Partition -DiskId $Path).DiskId
    $Disk = ( Get-Disk -Path $Path).Number
    $EbsVolumeId  = GetEBSVolumeId($Path)
    $Size =(Get-Disk -Path $Path).Size
    $DriveLetter,$VirtualDevice, $Partition = (GetDriveLetter($Path))
    $Device,$VolumeName = GetDeviceName($EbsVolumeId)
    $Disk = New-Object PSObject -Property @{
      Disk          = $Disk
      Partitions    = $Partition
      DriveLetter   = $DriveLetter
      EbsVolumeId   = $EbsVolumeId
      Device        = $Device
      VirtualDevice = $VirtualDevice
      VolumeName= $VolumeName
    }
 $Report += $Disk
}

$Report | Sort-Object Disk | Format-Table -AutoSize -Property Disk, Partitions,
 DriveLetter, EbsVolumeId, Device, VirtualDevice, VolumeName
```

Jalankan skrip sebagai berikut:

```
PS C:\> .\mapping.ps1
```

Berikut ini adalah contoh output untuk sebuah instans dengan volume root, dua volume EBS, dan dua volume penyimpanan instans.

```
Disk Partitions DriveLetter EbsVolumeId           Device    VirtualDevice VolumeName
---- ---------- ----------- -----------           ------    ------------- ----------
   0          1 C           vol-03683f1d861744bc7 /dev/sda1 root
   1          1 D           vol-082b07051043174b9 xvdb      N/A
   2          1 E           vol-0a4064b39e5f534a2 xvdc      N/A
   3          1 F           AWS-6AAD8C2AEEE1193F0 Ephemeral N/A           Temporary
 Storage
   4          1 G           AWS-13E7299C2BD031A28 Ephemeral N/A           Temporary
 Storage
```

Jika Anda tidak mengonfigurasi kredenSIAL untuk Tools for Windows PowerShell pada instance Windows, skrip tidak dapat mendapatkan ID volume EBS dan penggunaan N/A di `EbsVolumeId` kolom.

## Petakan disk NVMe ke volume

Anda dapat menggunakan Get-Disk perintah untuk memetakan nomor disk Windows ke volume Amazon EBS dan volume penyimpanan instans Amazon EC2.

```
PS C:\> Get-Disk
Number Friendly Name Serial Number                    HealthStatus
 OperationalStatus      Total Size Partition

                      Style
------ ------------- -------------                    ------------
 -----------------      ---------- ----------
3      NVMe Amazo... AWS6AAD8C2AEEE1193F0_00000001.   Healthy              Online
              279.4 GB MBR
4      NVMe Amazo... AWS13E7299C2BD031A28_00000001.   Healthy              Online
              279.4 GB MBR
2      NVMe Amazo... vol0a4064b39e5f534a2_00000001.   Healthy              Online
                8 GB MBR
0      NVMe Amazo... vol03683f1d861744bc7_00000001.   Healthy              Online
                30 GB MBR
```

```
1      NVMe Amazo... vol082b07051043174b9_00000001.   Healthy              Online
                8 GB MBR
```

Anda juga dapat menjalankan perintah ebsnvme-id untuk memetakan angka disk NVMe ke ID volume EBS dan nama perangkat.

```
PS C:\> C:\PROGRAMDATA\Amazon\Tools\ebsnvme-id.exe
Disk Number: 0
Volume ID: vol-03683f1d861744bc7
Device Name: sda1

Disk Number: 1
Volume ID: vol-082b07051043174b9
Device Name: xvdb

Disk Number: 2
Volume ID: vol-0a4064b39e5f534a2
Device Name: xvdc
```

## Memetakan disk non-NVMe pada instans Windows Amazon EC2 ke volume

Untuk instans yang diluncurkan dari AMI Windows yang menggunakan AWS driver PV atau Citrix PV, Anda dapat menggunakan hubungan yang dijelaskan di halaman ini untuk memetakan disk Windows Anda ke penyimpanan instans dan volume EBS. Topik ini menjelaskan cara melihat disk non-NVMe yang tersedia untuk sistem operasi Windows pada instans Anda. Ini juga menunjukkan cara memetakan disk non-NVMe tersebut ke volume Amazon EBS yang mendasarinya dan nama perangkat yang ditentukan untuk pemetaan perangkat blok yang digunakan oleh Amazon EC2.

> **Note**
>
> Jika Anda meluncurkan instance Jika AMI Windows Anda menggunakan driver Red Hat PV, Anda dapat memperbarui instance Anda untuk menggunakan driver Citrix. Untuk informasi selengkapnya, lihat the section called “Mutakhirkan driver PV”.

Topik


- Daftar disk non-NVMe
- Memetakan disk non-NVMe ke volume

## Daftar disk non-NVMe

Anda dapat menemukan disk pada instance Windows Anda menggunakan Manajemen Disk atau PowerShell.

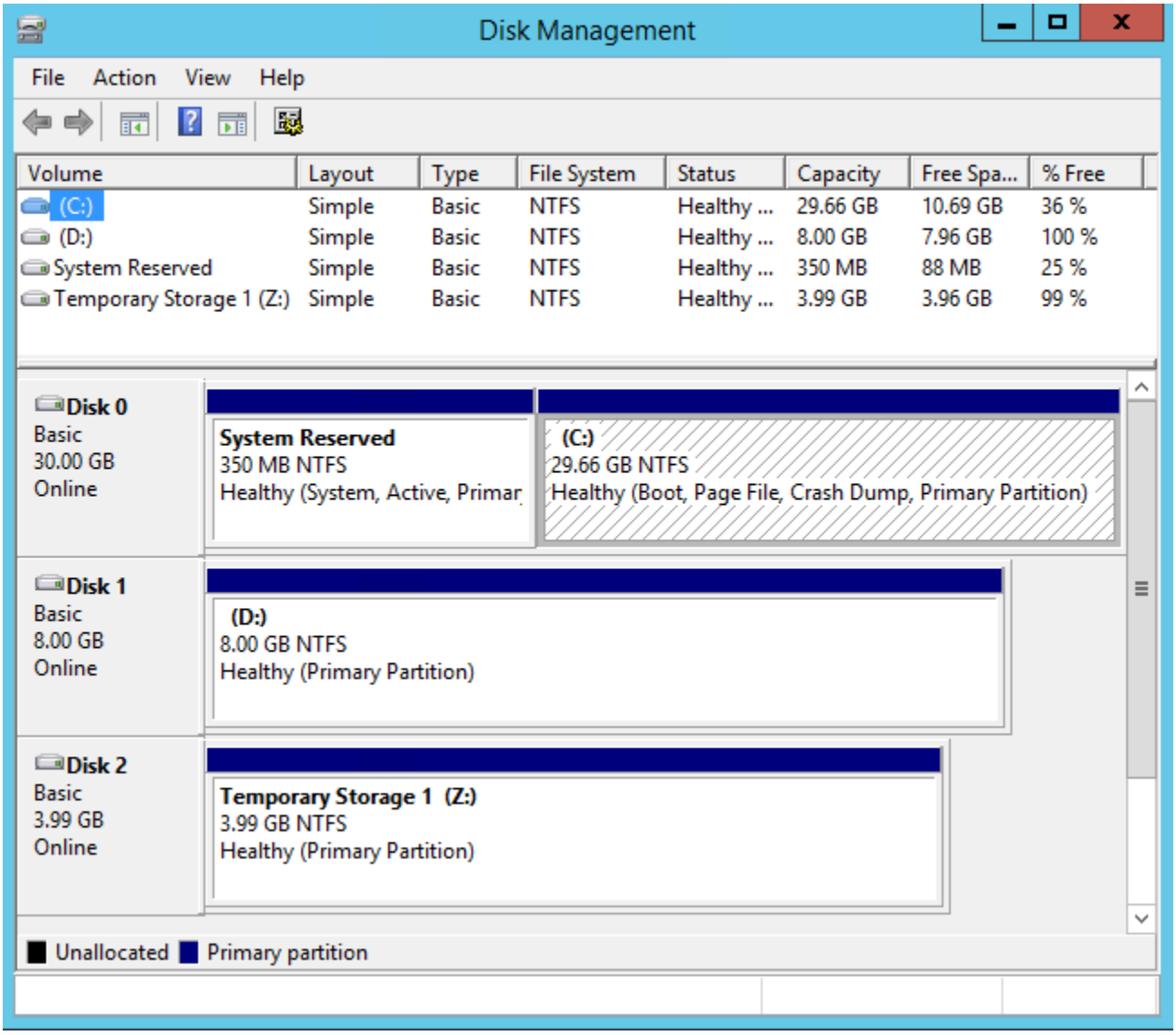
### Disk Management

Untuk menemukan disk di instans Windows Anda

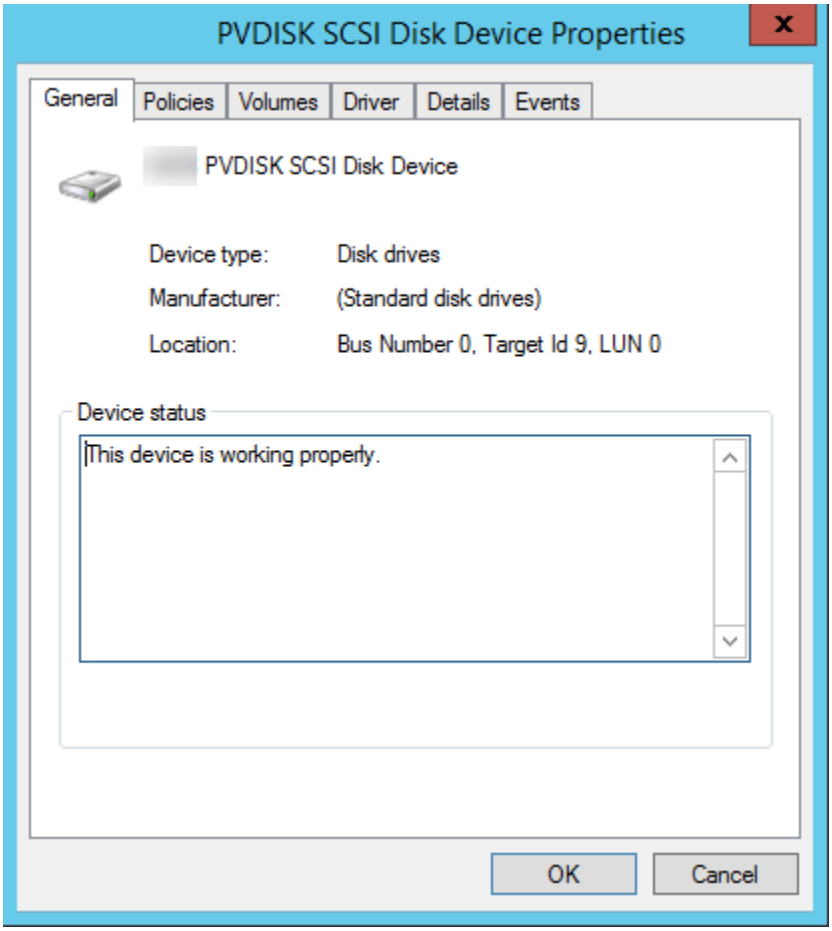
1. Masuk ke instans Windows menggunakan Remote Desktop. Untuk informasi selengkapnya, lihat Hubungkan ke instans Windows Anda menggunakan RDP.
2. Mulai utilitas Manajemen Disk.

   Pada bilah tugas, buka menu konteks (klik kanan) untuk logo Windows, lalu pilih Manajemen Disk.
3. Tinjau disk. Volume root adalah volume EBS yang dipasang sebagai `C:\`. Jika tidak ada disk lain yang ditampilkan, berarti Anda tidak menentukan volume tambahan saat membuat AMI atau meluncurkan instans.

   Berikut ini adalah instans yang menunjukkan disk yang tersedia jika Anda meluncurkan instans `m3.medium` dengan volume penyimpanan instans (Disk 2) dan volume EBS tambahan (Disk 1).




4. Buka menu konteks (klik kanan) untuk panel abu-abu berlabel Disk 1, lalu pilih Properties. Perhatikan nilai Lokasi dan cari dalam tabel di Memetakan disk non-NVMe ke volume. Misalnya, disk berikut memiliki lokasi Nomor Bus 0, Target Id 9, LUN 0. Menurut tabel untuk volume EBS, nama perangkat untuk lokasi ini adalah `xvdj`.




PowerShell

PowerShell Skrip berikut mencantumkan setiap disk dan nama dan volume perangkat yang sesuai.

Persyaratan dan batasan

- Memerlukan Windows Server 2012 atau yang lebih baru.
- Memerlukan kredensial untuk mendapatkan ID volume EBS. Anda dapat mengonfigurasi profil menggunakan Tools for PowerShell, atau melampirkan peran IAM ke instance.
- Tidak mendukung volume NVMe.
- Tidak mendukung disk dinamis.

Hubungkan ke instans Windows Anda dan jalankan perintah berikut untuk mengaktifkan PowerShell pelaksanaan skrip.

```
Set-ExecutionPolicy RemoteSigned
```

Salin skrip berikut dan simpan sebagai `mapping.ps1` di instans Windows Anda.

```
# List the disks
function Convert-SCSITargetIdToDeviceName {
  param([int]$SCSITargetId)
  If ($SCSITargetId -eq 0) {
    return "sda1"
  }
  $deviceName = "xvd"
  If ($SCSITargetId -gt 25) {
    $deviceName += [char](0x60 + [int]($SCSITargetId / 26))
  }
  $deviceName += [char](0x61 + $SCSITargetId % 26)
  return $deviceName
}

[string[]]$array1 = @()
[string[]]$array2 = @()
[string[]]$array3 = @()
[string[]]$array4 = @()

Get-WmiObject Win32_Volume | Select-Object Name, DeviceID | ForEach-Object {
  $array1 += $_.Name
  $array2 += $_.DeviceID
}

$i = 0
While ($i -ne ($array2.Count)) {
  $array3 += ((Get-Volume -Path $array2[$i] | Get-Partition | Get-
Disk).SerialNumber) -replace "_[^ ]*$" -replace "vol", "vol-"
  $array4 += ((Get-Volume -Path $array2[$i] | Get-Partition | Get-
Disk).FriendlyName)
  $i ++
}

[array[]]$array = $array1, $array2, $array3, $array4

Try {
  $InstanceId = Get-EC2InstanceMetadata -Category "InstanceId"
  $Region = Get-EC2InstanceMetadata -Category "Region" | Select-Object -
ExpandProperty SystemName
```

```
}
Catch {
  Write-Host "Could not access the instance Metadata using AWS Get-
EC2InstanceMetadata CMDLet.
Verify you have AWSPowershell SDK version '3.1.73.0' or greater installed and
 Metadata is enabled for this instance." -ForegroundColor Yellow
}
Try {
  $BlockDeviceMappings = (Get-EC2Instance -Region $Region -Instance
 $InstanceId).Instances.BlockDeviceMappings
  $VirtualDeviceMap = (Get-EC2InstanceMetadata -Category
 "BlockDeviceMapping").GetEnumerator() | Where-Object { $_.Key -ne "ami" }
}
Catch {
  Write-Host "Could not access the AWS API, therefore, VolumeId is not available.
Verify that you provided your access keys or assigned an IAM role with adequate
 permissions." -ForegroundColor Yellow
}

Get-disk | ForEach-Object {
  $DriveLetter = $null
  $VolumeName = $null
  $VirtualDevice = $null
  $DeviceName = $_.FriendlyName

  $DiskDrive = $_
  $Disk = $_.Number
  $Partitions = $_.NumberOfPartitions
  $EbsVolumeID = $_.SerialNumber -replace "_[^ ]*$" -replace "vol", "vol-"
  if ($Partitions -ge 1) {
    $PartitionsData = Get-Partition -DiskId $_.Path
    $DriveLetter = $PartitionsData.DriveLetter | Where-object { $_ -notin @("",
 $null) }
    $VolumeName = (Get-PSDrive | Where-Object { $_.Name -in
 @($DriveLetter) }).Description | Where-object { $_ -notin @("", $null) }
  }
  If ($DiskDrive.path -like "*PROD_PVDISK*") {
    $BlockDeviceName = Convert-SCSITargetIdToDeviceName((Get-WmiObject -Class
 Win32_Diskdrive | Where-Object { $_.DeviceID -eq ("\\.\PHYSICALDRIVE" +
 $DiskDrive.Number) }).SCSITargetId)
    $BlockDeviceName = "/dev/" + $BlockDeviceName
    $BlockDevice = $BlockDeviceMappings | Where-Object { $BlockDeviceName -like "*"
 + $_.DeviceName + "*" }
    $EbsVolumeID = $BlockDevice.Ebs.VolumeId
```

```
    $VirtualDevice = ($VirtualDeviceMap.GetEnumerator() | Where-Object { $_.Value -
eq $BlockDeviceName }).Key | Select-Object -First 1
  }
  ElseIf ($DiskDrive.path -like "*PROD_AMAZON_EC2_NVME*") {
    $BlockDeviceName = (Get-EC2InstanceMetadata -Category
 "BlockDeviceMapping")."ephemeral$((Get-WmiObject -Class Win32_Diskdrive | Where-
Object { $_.DeviceID -eq ("\\.\PHYSICALDRIVE" + $DiskDrive.Number) }).SCSIPort - 2)"
    $BlockDevice = $null
    $VirtualDevice = ($VirtualDeviceMap.GetEnumerator() | Where-Object { $_.Value -
eq $BlockDeviceName }).Key | Select-Object -First 1
  }
  ElseIf ($DiskDrive.path -like "*PROD_AMAZON*") {
    if ($DriveLetter -match '[^a-zA-Z0-9]') {
      $i = 0
      While ($i -ne ($array3.Count)) {
        if ($array[2][$i] -eq $EbsVolumeID) {
          $DriveLetter = $array[0][$i]
          $DeviceName = $array[3][$i]
        }
        $i ++
      }
    }
    $BlockDevice = ""
    $BlockDeviceName = ($BlockDeviceMappings | Where-Object { $_.ebs.VolumeId -eq
 $EbsVolumeID }).DeviceName
  }
  ElseIf ($DiskDrive.path -like "*NETAPP*") {
    if ($DriveLetter -match '[^a-zA-Z0-9]') {
      $i = 0
      While ($i -ne ($array3.Count)) {
        if ($array[2][$i] -eq $EbsVolumeID) {
          $DriveLetter = $array[0][$i]
          $DeviceName = $array[3][$i]
        }
        $i ++
      }
    }
    $EbsVolumeID = "FSxN Volume"
    $BlockDevice = ""
    $BlockDeviceName = ($BlockDeviceMappings | Where-Object { $_.ebs.VolumeId -eq
 $EbsVolumeID }).DeviceName
  }
  Else {
    $BlockDeviceName = $null
```

```
    $BlockDevice = $null
  }
  New-Object PSObject -Property @{
    Disk          = $Disk;
    Partitions    = $Partitions;
    DriveLetter   = If ($DriveLetter -eq $null) { "N/A" } Else { $DriveLetter };
    EbsVolumeId   = If ($EbsVolumeID -eq $null) { "N/A" } Else { $EbsVolumeID };
    Device        = If ($BlockDeviceName -eq $null) { "N/A" } Else
 { $BlockDeviceName };
    VirtualDevice = If ($VirtualDevice -eq $null) { "N/A" } Else { $VirtualDevice };
    VolumeName    = If ($VolumeName -eq $null) { "N/A" } Else { $VolumeName };
    DeviceName    = If ($DeviceName -eq $null) { "N/A" } Else { $DeviceName };
  }
} | Sort-Object Disk | Format-Table -AutoSize -Property Disk, Partitions,
 DriveLetter, EbsVolumeId, Device, VirtualDevice, DeviceName, VolumeName
```

Jalankan skrip sebagai berikut:

```
PS C:\> .\mapping.ps1
```

Berikut ini adalah output contoh.

```
Disk  Partitions  DriveLetter   EbsVolumeId            Device      VirtualDevice
 DeviceName              VolumeName
----  ----------  -----------   -----------            ------      -------------
 ----------              ----------
   0           1              C   vol-0561f1783298efedd   /dev/sda1   N/A
 NVMe Amazon Elastic B   N/A
   1           1              D   vol-002a9488504c5e35a   xvdb        N/A
 NVMe Amazon Elastic B   N/A
   2           1              E   vol-0de9d46fcc907925d   xvdc        N/A
 NVMe Amazon Elastic B   N/A
```

Jika Anda tidak memberikan kredenSIAL pada instance Windows, skrip tidak dapat memperoleh ID volume EBS dan penggunaan N/A di `EbsVolumeId` kolom.

## Memetakan disk non-NVMe ke volume

Driver perangkat blok untuk instans menetapkan nama volume aktual saat melakukan pemasangan volume.

Pemetaan

- Volume penyimpanan instans
- EBS volume

## Volume penyimpanan instans

Tabel berikut menjelaskan bagaimana driver PV dan PV Citrix memet AWS akan volume penyimpanan instans non-NVMe ke volume Windows. Jumlah volume penyimpanan instans yang tersedia ditentukan oleh tipe instans. Untuk informasi selengkapnya, lihat Batas volume penyimpanan instans untuk instans EC2.

| Lokasi | Nama perangkat |
|---|---|
| Nomor Bus 0, ID Target 78, LUN 0 | xvdca |
| Nomor Bus 0, ID Target 79, LUN 0 | xvdcb |
| Nomor Bus 0, ID Target 80, LUN 0 | xvdcc |
| Nomor Bus 0, ID Target 81, LUN 0 | xvdcd |
| Nomor Bus 0, ID Target 82, LUN 0 | xvdce |
| Nomor Bus 0, ID Target 83, LUN 0 | xvdcf |
| Nomor Bus 0, ID Target 84, LUN 0 | xvdcg |
| Nomor Bus 0, ID Target 85, LUN 0 | xvdch |
| Nomor Bus 0, ID Target 86, LUN 0 | xvdci |
| Nomor Bus 0, ID Target 87, LUN 0 | xvdcj |
| Nomor Bus 0, ID Target 88, LUN 0 | xvdck |
| Nomor Bus 0, ID Target 89, LUN 0 | xvdcl |

EBS volume

Tabel berikut menjelaskan bagaimana driver Citrix PV dan PV memet AWS akan volume EBS non-NVMe ke volume Windows.

| Lokasi | Nama perangkat |
| --- | --- |
| Nomor Bus 0, ID Target 0, LUN 0 | /dev/sda1 |
| Nomor Bus 0, ID Target 1, LUN 0 | xvdb |
| Nomor Bus 0, ID Target 2, LUN 0 | xvdc |
| Nomor Bus 0, ID Target 3, LUN 0 | xvdd |
| Nomor Bus 0, ID Target 4, LUN 0 | xvde |
| Nomor Bus 0, ID Target 5, LUN 0 | xvdf |
| Nomor Bus 0, ID Target 6, LUN 0 | xvdg |
| Nomor Bus 0, ID Target 7, LUN 0 | xvdh |
| Nomor Bus 0, ID Target 8, LUN 0 | xvdi |
| Nomor Bus 0, ID Target 9, LUN 0 | xvdj |
| Nomor Bus 0, ID Target 10, LUN 0 | xvdk |
| Nomor Bus 0, ID Target 11, LUN 0 | xvdl |
| Nomor Bus 0, ID Target 12, LUN 0 | xvdm |
| Nomor Bus 0, ID Target 13, LUN 0 | xvdn |
| Nomor Bus 0, ID Target 14, LUN 0 | xvdo |
| Nomor Bus 0, ID Target 15, LUN 0 | xvdp |
| Nomor Bus 0, ID Target 16, LUN 0 | xvdq |
| Nomor Bus 0, ID Target 17, LUN 0 | xvdr |

| Lokasi | Nama perangkat |
| --- | --- |
| Nomor Bus 0, ID Target 18, LUN 0 | xvds |
| Nomor Bus 0, ID Target 19, LUN 0 | xvdt |
| Nomor Bus 0, ID Target 20, LUN 0 | xvdu |
| Nomor Bus 0, ID Target 21, LUN 0 | xvdv |
| Nomor Bus 0, ID Target 22, LUN 0 | xvdw |
| Nomor Bus 0, ID Target 23, LUN 0 | xvdx |
| Nomor Bus 0, ID Target 24, LUN 0 | xkertas |
| Nomor Bus 0, ID Target 25, LUN 0 | xvdz |

# Pencegahan penulisan robek pada instans Amazon EC2 Linux

> **Note**
>
> Pencegahan penulisan robek hanya didukung untuk instans Linux.

Pencegahan penulisan robek adalah fitur penyimpanan blok yang dirancang AWS untuk meningkatkan kinerja beban kerja database I/O-intensive relasional Anda dan mengurangi latensi tanpa berdampak negatif pada ketahanan data. Basis data relasional yang menggunakan InnoDB atau XtraDB sebagai mesin basis data, seperti MySQL dan MariaDB, akan mendapat manfaat dari pencegahan penulisan yang rusak.

Biasanya, basis data relasional yang menggunakan halaman yang lebih besar dari daya gagal atomisitas perangkat penyimpanan menggunakan mekanisme pencatatan data untuk melindungi dari tumpang tindih. MariaDB dan MySQL menggunakan file buffer doublewrite untuk mencatat data sebelum menulisnya ke tabel data. Jika terjadi penulisan yang tidak lengkap atau sobek, sebagai akibat dari kerusakan sistem operasi atau kehilangan daya selama transaksi penulisan, basis data dapat memulihkan data dari buffer penulisan ganda. I/O Overhead tambahan yang terkait dengan penulisan ke buffer doublewrite berdampak pada kinerja database dan latensi aplikasi, dan

mengurangi jumlah transaksi yang dapat diproses per detik. Untuk informasi selengkapnya tentang buffer doublewrite, lihat dokumentasi MariaDB dan MySQL.

Dengan pencegahan tumpang tindih, data ditulis ke penyimpanan dalam transaksi tulis semua atau tidak sama sekali, yang menghilangkan kebutuhan untuk menggunakan buffer doublewrite. Hal ini mencegah data parsial, atau tumpang tindih, ditulis ke penyimpanan jika terjadi crash sistem operasi atau kehilangan daya selama transaksi tulis. Jumlah transaksi yang diproses per detik dapat ditingkatkan hingga 30 persen, dan latensi penulisan dapat dikurangi hingga 50 persen, tanpa mengorbankan ketahanan beban kerja Anda.

Harga

Tidak ada biaya tambahan untuk menggunakan pencegahan tumpang tindih.

Daftar Isi


- Ukuran blok untuk pencegahan penulisan robek di Amazon EC2
- Persyaratan untuk menggunakan pencegahan penulisan robek di Amazon EC2
- Periksa dukungan instans Amazon EC2 untuk pencegahan penulisan robek
- Konfigurasikan beban kerja Anda di Amazon EC2 untuk pencegahan penulisan robek


## Ukuran blok untuk pencegahan penulisan robek di Amazon EC2

Pencegahan tumpang tindih mendukung operasi penulisan untuk blok data 4 KiB, 8 KiB, dan 16 KiB. Alamat blok logis awal blok data (LBA) harus disejajarkan dengan ukuran batas blok masing-masing yaitu 4 KiB, 8 KiB, atau 16 KiB. Misalnya, untuk operasi penulisan 16 KiB, LBA awal blok data harus disejajarkan dengan ukuran batas blok 16 KiB.

Tabel berikut menunjukkan dukungan di seluruh tipe penyimpanan dan instans.

| | 4 KiB blok | 8 KiB blok | 16 KiB blok |
|---|---|---|---|
| Volume penyimpanan instans | Semua instance NVMe menyimpan volume yang dilampirkan ke I-family instans generasi saat ini. | Instans i4i, iM4GN, IS4Gen, i7i, i7ie, i8G, dan i8GE didukung oleh Nitro SSD. AWS | |

| | 4 KiB blok | 8 KiB blok | 16 KiB blok |
|---|---|---|---|
| Volume Amazon EBS | Semua volume Amazon EBS dilampirkan ke Nitro-based inst ans. | | |

Untuk mengonfirmasi apakah instans dan volume Anda mendukung pencegahan tumpang tindih, lakukan kueri untuk memeriksa apakah instans mendukung pencegahan penulisan sobek dan detail lainnya, seperti ukuran blok dan batas yang didukung. Untuk informasi selengkapnya, lihat Periksa dukungan instans Amazon EC2 untuk pencegahan penulisan robek.

## Persyaratan untuk menggunakan pencegahan penulisan robek di Amazon EC2

Agar pencegahan penulisan robek berfungsi dengan baik, I/O operasi harus memenuhi persyaratan ukuran, penyelarasan, dan batas, seperti yang ditentukan dalam NTWBU bidang NTWPUNTWGU,,. Anda harus mengonfigurasi sistem operasi Anda untuk memastikan bahwa subsistem penyimpanan tertentu (sistem file, LVM, RAID, dll) tidak mengubah I/O properti di tumpukan penyimpanan, termasuk penggabungan blok, pemisahan, atau relokasi alamat blok, sebelum dikirim ke perangkat.

Pencegahan tumpang tinding telah diuji dengan konfigurasi berikut:

- Tipe instans dan tipe penyimpanan yang mendukung ukuran blok yang diperlukan.
- Amazon Linux 2 dengan kernel versi 5.10 atau yang lebih baru.
- ext4 dengan `bigalloc` diaktifkan dan ukuran klaster 16 KiB, serta utilitas ext4 terbaru (e2fsprogs 1.46.5 atau yang lebih baru).
- mode akses file `O_DIRECT` untuk melakukan bypass cache buffer kernel Linux.

> **Note**
>
> Anda tidak perlu menonaktifkan I/O penggabungan untuk beban kerja MySQL dan MariaDB.

# Periksa dukungan instans Amazon EC2 untuk pencegahan penulisan robek

Untuk mengonfirmasi apakah instance dan volume Anda mendukung pencegahan penulisan robek, dan untuk melihat data khusus vendor namespace NVMe yang berisi informasi pencegahan penulisan yang robek, gunakan perintah berikut.

```
$ sudo nvme id-ns -v device_name
```

> **Note**
>
> Perintah ini mengembalikan informasi khusus vendor dalam heksa dengan interpretasi ASCII. Anda mungkin perlu membuat alat yang mirip dengan `ebsnvme-id`, ke dalam aplikasi Anda yang dapat membaca dan mengurai output.

Misalnya, perintah berikut mengembalikan data spesifik vendor namespace NVMe yang berisi informasi pencegahan penulisan yang robek untuk. `/dev/nvme1n1`

```
$ sudo nvme id-ns -v /dev/nvme1n1
```

Jika instance dan volume Anda mendukung pencegahan penulisan robek, ia mengembalikan informasi pencegahan penul AWS isan robek berikut dalam data spesifik vendor namespace NVMe.

> **Note**
>
> Byte dalam tabel berikut mewakili offset dalam byte dari awal data spesifik vendor namespace NVMe.

| Byte | Deskripsi |
| --- | --- |
| `0:31` | Nama titik pemasangan lampiran perangkat, misalnya `/dev/xvda` . Anda memberikan ini selama permintaan lampiran volume dan dapat digunakan oleh instans Amazon EC2 untuk membuat symlink ke perangkat blok NVMe (`nvmeXn1`). |

| Byte | Deskripsi |
| --- | --- |
| 32:63 | ID volume. Misalnya, `vol01234567890abcdef` . Bidang ini dapat digunakan untuk memetakan perangkat NVMe ke volume yang terlampir. |
| 64:255 | Terpesan untuk digunakan di masa mendatang. |
| 256:257 | Ukuran Unit Pencegahan Tumpang Tindih Namespace (NTWPU). Bidang ini menunjukkan ukuran spesifik ruang nama dari operasi penulisan yang dijamin akan ditulis secara atomik ke NVM selama terjadi kegagalan daya atau kondisi kesalahan. Bidang ini ditentukan dalam blok logis yang direpresentasikan dalam nilai berbasis nol. |
| 258:259 | Ukuran Perincian Pencegahan Tumpang Tindih Namespace (NTWPG). Bidang ini menunjukkan kenaikan ukuran spesifik namespace di bawah `NTWPU` operasi tulis yang dijamin akan ditulis secara atomik ke NVM selama kegagalan daya atau kondisi kesalahan. Artinya, ukuran harus `NTWPG * n <= NTWPU`, yaitu n adalah bilangan bulat positif. Operasi tulis offset LBA juga harus diselaraskan dengan bidang ini. Bidang ini ditentukan dalam blok logis yang direpresentasikan dalam nilai berbasis nol. |
| 260:263 | Ukuran Batas Pencegahan Tumpang Tindih Namespace (NTWPB). Bidang ini menunjukkan ukuran batas atom untuk namespace ini untuk nilai `NTWPU`. Menulis ke namespace ini yang melintasi batas atom tidak dijamin akan ditulis secara atomik ke NVM selama kondisi kegagalan daya atau kesalahan. Nilai `0h` menunjukkan bahwa tidak ada batas atomis untuk kegagalan daya atau kondisi kesalahan. Semua nilai lainnya menentukan ukuran dalam hal blok logis menggunakan pengodean yang sama dengan bidang `NTWPU`. |

## Konfigurasikan beban kerja Anda di Amazon EC2 untuk pencegahan penulisan robek

Pencegahan tumpang tindih diaktifkan secara default pada tipe instans yang didukung dengan volume yang didukung. Anda tidak perlu mengonfigurasi pengaturan tambahan apa pun untuk mengaktifkan pencegahan penulisan robek.

> **Note**
>
> Beban kerja yang tidak mendukung pencegahan penulisan robek tidak berdampak pada kinerja. Anda tidak perlu membuat perubahan apa pun untuk beban kerja ini.
> Beban kerja yang mendukung pencegahan penulisan robek, tetapi tidak dikonfigurasi untuk menggunakannya, terus menggunakan buffer doublewrite dan tidak menerima manfaat kinerja apa pun.

Untuk mengonfigurasi tumpukan perangkat lunak MySQL atau MariaDB Anda guna menonaktifkan buffer penulisan ganda dan menggunakan pencegahan tumpang tindih, selesaikan langkah-langkah berikut:

1. Konfigurasikan volume Anda untuk menggunakan sistem file ext4 dengan BigAlloc opsi dan atur ukuran cluster menjadi 4 KiB, 8 KiB, atau 16 KiB. Menggunakan BigAlloc dengan ukuran cluster 4 KiB, 8 KiB, atau 16 KiB memastikan bahwa sistem file mengalokasikan file yang sejajar dengan batas masing-masing.

   ```
   $  mkfs.ext4 -O bigalloc -C 4096|8192|16384 device_name
   ```

   > **Note**
   >
   > Untuk MySQL dan MariaDB, Anda harus menggunakan `-C 16384` untuk mencocokkan ukuran halaman basis data. Mengatur perincian alokasi ke nilai selain kelipatan ukuran halaman dapat mengakibatkan alokasi yang mungkin tidak cocok dengan batasan pencegahan tumpang tindih pada perangkat penyimpanan.

   Misalnya:

   ```
   $  mkfs.ext4 -O bigalloc -C 16384 /dev/nvme1n1
   ```

2. Konfigurasikan InnoDB untuk menggunakan metode flush `O_DIRECT` dan matikan doublewrite InnoDB. Gunakan editor teks pilihan Anda untuk membuka `/etc/my.cnf`, dan perbarui parameter `innodb_flush_method` dan `innodb_doublewrite` sebagai berikut:

   ```
   innodb_flush_method=O_DIRECT
   ```

```
innodb_doublewrite=0
```

**Important**

Jika Anda menggunakan Logical Volume Manager (LVM) atau lapisan virtualisasi penyimpanan lainnya, pastikan offset awal volume disejajarkan pada kelipatan 16 KiB. Hal ini relatif terhadap penyimpanan NVMe yang mendasarinya untuk memperhitungkan header metadata dan superblock yang digunakan oleh lapisan virtualisasi penyimpanan. Jika Anda menambahkan offset ke volume fisik LVM, hal ini dapat menyebabkan ketidakselarasan antara alokasi sistem file dan offset perangkat NVMe, yang akan membatalkan pencegahan penulisan robek. Untuk informasi selengkapnya, lihat `--dataalignmentoffset` di halaman manual Linux.

# Snapshot Amazon EBS berbasis Windows VSS yang konsisten dengan aplikasi

Anda dapat mengambil snapshot yang konsisten dengan aplikasi dari semua volume Amazon EBS yang dilampirkan ke instans Windows Amazon EC2 menggunakan Run Command. AWS Systems Manager Proses snapshot menggunakan Windows Volume Shadow Copy Service (VSS) untuk mengambil cadangan aplikasi tingkat volume EBS. VSS-aware Snapshot mencakup data dari transaksi yang tertunda antara aplikasi ini dan disk. Anda tidak perlu mematikan instans atau memutusnya saat Anda perlu mencadangkan semua volume yang terlampir.

Tidak ada biaya tambahan untuk menggunakan snapshot EBS berbasis VSS. Anda hanya perlu membayar snapshot EBS yang dibuat oleh proses pencadangan. Untuk informasi selengkapnya, lihat Bagaimana saya ditagih untuk snapshot Amazon EBS saya?

**Note**

Snapshot berbasis Windows VSS yang konsisten aplikasi hanya didukung dengan instans Windows.

Daftar Isi


- Apa itu VSS?

- Cara kerja solusi snapshot Amazon EBS berbasis VSS
- Prasyarat untuk membuat snapshot EBS berbasis Windows VSS
- Buat snapshot EBS berbasis VSS untuk instans Windows EC2 Anda
- Memecahkan masalah snapshot EBS berbasis Windows VSS
- Gunakan AWS Solusi VSS untuk memulihkan data untuk instance Anda
- AWS riwayat versi VSS solution

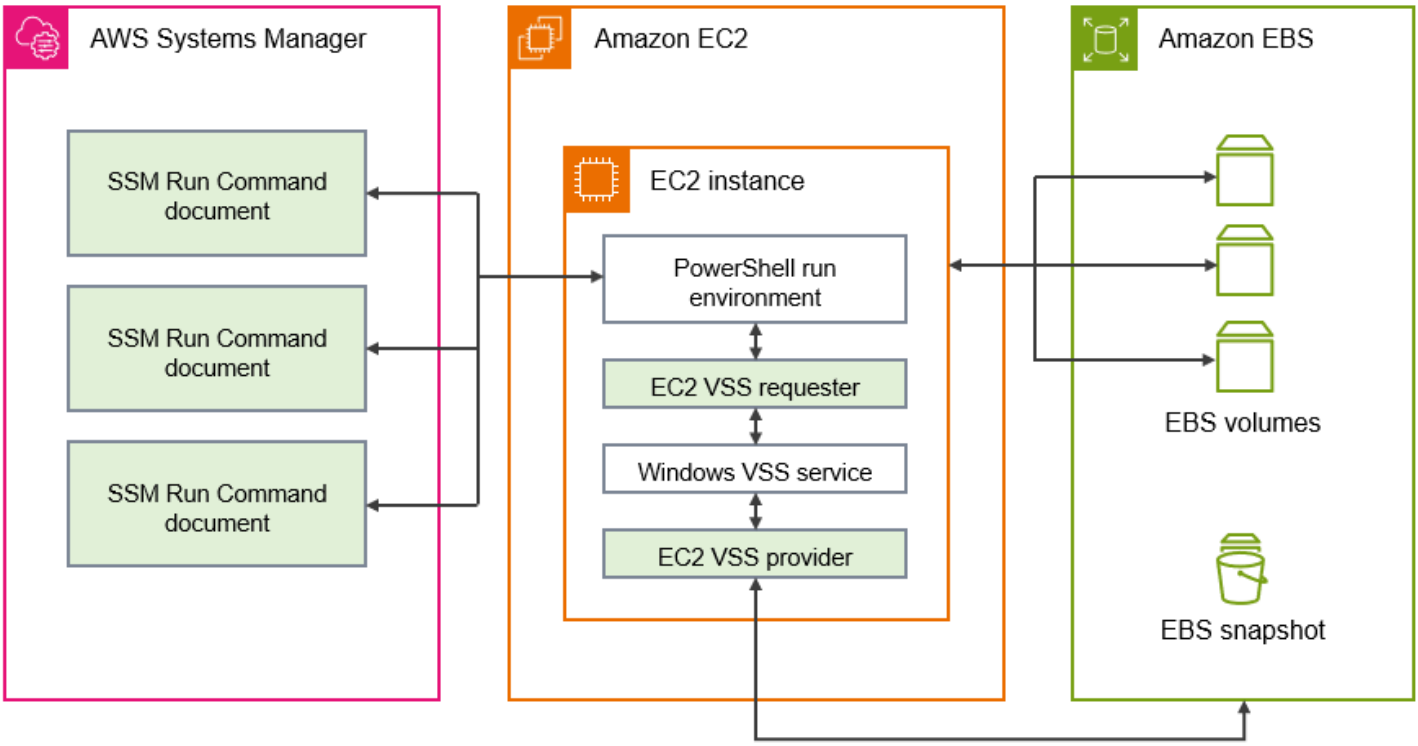

## Apa itu VSS?

Volume Shadow Copy Service (VSS) adalah teknologi cadangan dan pemulihan yang disertakan dalam Microsoft Windows. Layanan ini dapat membuat salinan cadangan, atau snapshot dari file komputer atau volume saat sedang digunakan. Untuk informasi selengkapnya, lihat Layanan Salinan Snapshot Volume.

Untuk membuat snapshot yang konsisten dengan aplikasi, komponen perangkat lunak berikut ini terlibat.

- Layanan VSS — Bagian dari sistem operasi Windows
- Pemohon VSS — Perangkat lunak yang meminta pembuatan salinan bayangan
- Penulis VSS — Biasanya disediakan sebagai bagian dari aplikasi, seperti SQL Server, untuk memastikan set data yang konsisten untuk dicadangkan
- Penyedia VSS — Komponen yang membuat salinan bayangan dari volume yang mendasarinya

Solusi snapshot Amazon EBS berbasis Windows VSS terdiri dari beberapa dokumen Run Command System Manager (SSM) yang memfasilitasi pembuatan cadangan, dan paket Distributor Manajer Sistem, yang disebut `AwsVssComponents`, yang mencakup pemohon EC2 VSS dan penyedia EC2 VSS. Paket `AwsVssComponents` harus diinstal pada instans Windows EC2 untuk mengambil snapshot volume EBS yang konsisten dengan aplikasi. Diagram berikut menggambarkan hubungan antara komponen perangkat lunak ini.




## Cara kerja solusi snapshot Amazon EBS berbasis VSS

Proses pengambilan skrip snapshot EBS berbasis VSS yang konsisten dengan aplikasi terdiri dari langkah-langkah berikut.

1. Selesaikan Prasyarat untuk membuat snapshot EBS berbasis Windows VSS.
2. Masukkan parameter untuk dokumen SSM `AWSEC2-VssInstallAndSnapshot` dan jalankan dokumen ini menggunakan Run Command. Untuk informasi selengkapnya, lihat Jalan AWSEC2-VssInstallAndSnapshot kan dokumen perintah (disarankan).
3. Layanan Windows VSS pada instans Anda mengoordinasikan semua I/O operasi yang sedang berlangsung untuk menjalankan aplikasi.
4. Sistem membilas semua I/O buffer dan sementara menjeda semua operasi. I/O Jeda bertahan, paling sering, sepuluh detik.
5. Selama jeda, sistem membuat snapshot dari semua volume yang terlampir pada instans.
6. Jeda diangkat dan I/O melanjutkan operasi.
7. Sistem menambahkan semua snapshot yang baru dibuat ke daftar snapshot EBS. Sistem menandai semua snapshot EBS berbasis VSS yang berhasil dibuat oleh proses ini dengan. AppConsistent:true

8. Jika Anda perlu memulihkan dari snapshot, Anda dapat menggunakan proses EBS standar untuk membuat volume dari snapshot, atau Anda dapat memulihkan semua volume ke instans dengan menggunakan skrip contoh, seperti yang dijelaskan dalam Gunakan AWS Solusi VSS untuk memulihkan data untuk instance Anda.

## Prasyarat untuk membuat snapshot EBS berbasis Windows VSS

Anda dapat membuat snapshot EBS berbasis VSS dengan System Manager Run Command AWS Backup, atau Amazon Data Lifecycle Manager. Prasyarat berikut berlaku untuk semua solusi.

Persyaratan sistem

Pastikan instans Windows EC2 Anda memenuhi semua persyaratan sistem untuk membuat snapshot berbasis VSS, termasuk versi sistem operasi Windows yang didukung, kerangka kerja.NET, PowerShell AWS Tools for Windows PowerShell, dan Agen. AWS Systems Manager

Izin IAM

Peran IAM yang dilampirkan ke instans Windows Amazon EC2 Anda harus memiliki izin untuk membuat snapshot yang konsisten dengan aplikasi dengan VSS. Untuk memberikan izin yang diperlukan, Anda dapat melampirkan kebijakan ter `AWSEC2VssSnapshotPolicy` kelola ke profil instans Anda.

Komponen VSS

Untuk membuat snapshot yang konsisten aplikasi pada sistem operasi Windows, paket `AwsVssComponents` harus diinstal pada instans. Paket ini berisi Agen VSS EC2 pada instans yang berfungsi sebagai pemohon VSS, dan penyedia VSS EC2 untuk volume EBS.

### Persyaratan sistem

Instal Agen Manajer Sistem

VSS diatur oleh Agen Manajer Sistem menggunakan. PowerShell Pastikan Anda telah menginstal SSM Agent versi `3.0.502.0` atau versi yang lebih baru pada instans EC2 Anda. Jika Anda sudah menggunakan SSM Agent versi lama, perbarui menggunakan Run Command. Untuk informasi selengkapnya, lihat Menyiapkan Systems Manager untuk instans Amazon EC2 dan Bekerja dengan SSM Agent di instans Amazon EC2 untuk Windows Server di Panduan Pengguna AWS Systems Manager .

Persyaratan instans Windows Amazon EC2

Snapshot EBS berbasis VSS didukung untuk instans yang menjalankan Windows Server 2016 dan yang lebih baru.

Versi .NET Framework

Paket `AwsVssComponents` membutuhkan .NET Framework versi `4.6` atau yang lebih baru. Versi sistem operasi Windows sebelum Windows Server 2016 default ke versi sebelumnya dari .NET Framework. Jika instance Anda menggunakan versi sebelumnya dari.NET Framework, Anda harus menginstal versi `4.6` atau versi yang lebih baru menggunakan Pemutakhiran Windows.

AWS Tools for Windows PowerShell versi

Pastikan instans Anda menjalankan AWS Tools for Windows PowerShell versi `3.3.48.0` atau yang lebih baru. Untuk memeriksa versi Anda, jalankan perintah berikut di PowerShell terminal pada instance.

```
C:\> Get-AWSPowerShellVersion
```

Jika Anda perlu memperbarui AWS Tools for Windows PowerShell instans Anda, lihat Menginstal AWS Tools for Windows PowerShell di Panduan Alat AWS untuk PowerShell Pengguna.

PowerShell Versi Windows

Pastikan instans Anda menjalankan versi PowerShell utama Windows`3`,`4`, atau`5`. Untuk memeriksa versi Anda, jalankan perintah berikut di PowerShell terminal pada instance.

```
C:\> $PSVersionTable.PSVersion
```

PowerShell mode bahasa

Pastikan instans Anda memiliki mode PowerShell bahasa yang disetel ke`FullLanguage`. Untuk informasi selengkapnya, lihat about_ Language_Modes di dokumentasi Microsoft.

## Menggunakan kebijakan terkelola IAM untuk memberikan izin untuk snapshot berbasis VSS

Kebijakan AWSEC2VssSnapshotPolicy terkelola memungkinkan Systems Manager untuk melakukan tindakan berikut pada instans Windows Anda:

- Buat dan beri tag snapshot EBS
- Membuat dan menandai Amazon Machine Images (AMIs)
- Lampirkan metadata, seperti ID perangkat, ke tag snapshot default yang dibuat VSS.

Topik ini mencakup detail izin untuk kebijakan terkelola VSS, dan cara melampirkannya ke peran IAM profil instans EC2 Anda.

Daftar Isi


- AWSEC2VssSnapshotPolicyrincian kebijakan terkelola
- Lampirkan kebijakan terkelola snapshot VSS ke peran profil instans Anda


## AWSEC2VssSnapshotPolicyrincian kebijakan terkelola

Kebijakan AWS terkelola adalah kebijakan mandiri yang disediakan Amazon untuk AWS pelanggan. AWS kebijakan terkelola dirancang untuk memberikan izin untuk kasus penggunaan umum. Anda tidak dapat mengubah izin yang ditentukan dalam kebijakan AWS terkelola. Namun, Anda dapat menyalin kebijakan dan menggunakannya sebagai dasar untuk kebijakan yang dikelola pelanggan yang khusus untuk kasus penggunaan Anda.

Untuk informasi selengkapnya tentang kebijakan AWS AWS terkelola, lihat kebijakan terkelola di Panduan Pengguna IAM.

Untuk menggunakan kebijakan AWSEC2VssSnapshotPolicyterkelola, Anda dapat melampirkannya ke peran IAM yang dilampirkan ke Instans Windows EC2 Anda. Kebijakan ini memungkinkan solusi EC2 VSS untuk membuat dan menambahkan tag ke Amazon Machine Images (AMIs) dan Snapshots EBS. Untuk melampirkan kebijakan, lihatLampirkan kebijakan terkelola snapshot VSS ke peran profil instans Anda.

### Izin diberikan oleh AWSEC2VssSnapshotPolicy

AWSEC2VssSnapshotPolicyKebijakan ini mencakup izin Amazon EC2 berikut untuk mengizinkan Amazon EC2 membuat dan mengelola snapshot VSS atas nama Anda. Anda dapat melampirkan kebijakan terkelola ini ke peran profil instans IAM yang Anda gunakan untuk instans Windows EC2 Anda.

- ec2: CreateTags — Tambahkan tag ke snapshot EBS dan untuk membantu mengidentifikasi dan AMIs mengkategorikan sumber daya.

- ec2: DescribeInstanceAttribute — Ambil volume EBS dan pemetaan perangkat blok terkait yang dilampirkan ke instance target.
- ec2: CreateSnapshots — Buat snapshot volume EBS.
- ec2: CreateImage — Buat AMI dari instans EC2 yang sedang berjalan.
- ec2: DescribeImages — Ambil informasi untuk EC2 AMIs dan snapshot.
- ec2: DescribeSnapshots — Tentukan waktu pembuatan dan status snapshot untuk memverifikasi konsistensi aplikasi.

> **Note**
>
> Untuk melihat detail izin untuk kebijakan ini, lihat AWSEC2VssSnapshotPolicydi Referensi Kebijakan AWS Terkelola.

Merampingkan izin untuk kasus penggunaan tertentu - lanjutan

Kebijakan `AWSEC2VssSnapshotPolicy` terkelola menyertakan izin untuk semua cara Anda dapat membuat snapshot berbasis VSS. Anda dapat membuat kebijakan khusus yang hanya menyertakan izin yang Anda perlukan.

Kasus penggunaan: Buat AMI, Kasus penggunaan: Gunakan AWS Backup layanan

Jika Anda secara eksklusif menggunakan `CreateAmi` opsi, atau jika Anda membuat snapshot berbasis VSS hanya melalui AWS Backup layanan, maka Anda dapat merampingkan pernyataan kebijakan sebagai berikut.

- Menghilangkan pernyataan kebijakan yang diidentifikasi oleh pernyataan berikut IDs (SIDs):
  - `CreateSnapshotsWithTag`
  - `CreateSnapshotsAccessInstance`
  - `CreateSnapshotsAccessVolume`
- Sesuaikan `CreateTagsOnResourceCreation` pernyataan sebagai berikut:
  - Hapus `arn:aws:ec2:*:*:snapshot/*` dari sumber daya.
  - Hapus `CreateSnapshots` dari `ec2:CreateAction` kondisi.
- Sesuaikan `CreateTagsAfterResourceCreation` pernyataan untuk dihapus `arn:aws:ec2:*:*:snapshot/*` dari sumber daya.

- Sesuaikan `DescribeImagesAndSnapshots` pernyataan untuk dihapus `ec2:DescribeSnapshots` dari tindakan pernyataan.

Kasus penggunaan: Hanya snapshot

Jika Anda tidak menggunakan `CreateAmi` opsi, maka Anda dapat merampingkan pernyataan kebijakan sebagai berikut.

- Menghilangkan pernyataan kebijakan yang diidentifikasi oleh pernyataan berikut IDs (SIDs):
  - `CreateImageAccessInstance`
  - `CreateImageWithTag`
- Sesuaikan `CreateTagsOnResourceCreation` pernyataan sebagai berikut:
  - Hapus `arn:aws:ec2:*:*:image/*` dari sumber daya.
  - Hapus `CreateImage` dari `ec2:CreateAction` kondisi.
- Sesuaikan `CreateTagsAfterResourceCreation` pernyataan untuk dihapus `arn:aws:ec2:*:*:image/*` dari sumber daya.
- Sesuaikan `DescribeImagesAndSnapshots` pernyataan untuk dihapus `ec2:DescribeImages` dari tindakan pernyataan.

> **Note**
>
> Untuk memastikan bahwa kebijakan khusus Anda berjalan seperti yang diharapkan, kami sarankan Anda meninjau dan memasukkan pembaruan pada kebijakan terkelola secara berkala.

Lampirkan kebijakan terkelola snapshot VSS ke peran profil instans Anda

Untuk memberikan izin snapshot berbasis VSS untuk instans EC2 Windows, Anda dapat melampirkan kebijakan AWSEC2VssSnapshotPolicyterkelola ke peran profil instans Anda sebagai berikut. Penting untuk memastikan bahwa instans Anda memenuhi semuaPersyaratan sistem.

> **Note**
>
> Untuk menggunakan kebijakan terkelola, instans Anda harus memiliki versi `AwsVssComponents` paket `2.3.1` atau yang lebih baru diinstal. Untuk riwayat versi, lihatAwsVssComponents versi paket.

1. Buka konsol IAM di https://console.aws.amazon.com/iam/.
2. Di panel navigasi, pilih Peran untuk melihat daftar peran IAM yang dapat Anda akses.
3. Pilih tautan nama peran untuk peran yang dilampirkan ke instance Anda. Ini membuka halaman detail peran.
4. Untuk melampirkan kebijakan terkelola, pilih Tambahkan izin, yang terletak di sudut kanan atas panel daftar. Kemudian pilih Lampirkan kebijakan dari daftar dropdown.
5. Untuk merampingkan hasil, masukkan nama kebijakan di bilah pencarian (`AWSEC2VssSnapshotPolicy`).
6. Pilih kotak centang di samping nama kebijakan yang akan dilampirkan, lalu pilih Tambahkan izin.

## Kelola paket komponen VSS untuk snapshot EBS berbasis Windows VSS

Sebelum Anda membuat snapshot EBS berbasis VSS, pastikan Anda memiliki versi terbaru dari paket komponen VSS yang diinstal pada instans Windows Anda. Ada beberapa cara yang dapat Anda instal `AwsVssComponents` paket ke instance yang ada, sebagai berikut:

- (Direkomendasikan) Jalan AWSEC2-VssInstallAndSnapshot kan dokumen perintah (disarankan). Operasi ini secara otomatis menginstal atau memperbarui jika diperlukan setiap kali dijalankan.
- Instal komponen VSS secara manual pada instance EC2 Windows.
- Memperbarui paket komponen VSS pada instans Windows EC2 Anda.

Anda juga dapat membuat AMI dengan EC2 Image Builder yang menggunakan komponen terkelola `aws-vss-components-windows` guna menginstal paket `AwsVssComponents` untuk gambar. Komponen yang dikelola menggunakan AWS Systems Manager Distributor untuk menginstal paket. Setelah Image Builder membuat gambar, setiap instans yang Anda luncurkan dari AMI terkait akan menginstal paket VSS di dalamnya. Untuk informasi selengkapnya tentang cara membuat AMI dengan paket VSS yang diinstal, lihat Komponen terkelola paket distributor untuk Windows di Panduan Pengguna EC2 Image Builder.

Daftar Isi


- Instal komponen VSS secara manual pada instance EC2 Windows
- Memperbarui paket komponen VSS pada instans Windows EC2 Anda


## Instal komponen VSS secara manual pada instance EC2 Windows

Instans EC2 Windows Anda harus memiliki komponen VSS yang diinstal sebelum Anda dapat membuat snapshot yang bersifat konsisten aplikasi dengan Systems Manager. Jika Anda tidak menjalankan dokumen perintah `AWSEC2-VssInstallAndSnapshot` untuk menginstal atau memperbarui paket secara otomatis setiap kali Anda membuat snapshot yang bersifat konsisten aplikasi, Anda harus menginstal paket secara manual.

Anda juga harus menginstal secara manual jika Anda berencana untuk menggunakan salah satu metode berikut guna membuat snapshot yang bersifat konsisten aplikasi dari instans EC2.

- Buat snapshot VSS menggunakan AWS Backup
- Membuat snapshot VSS menggunakan Amazon Data Lifecycle Manager

Jika Anda perlu melakukan penginstalan manual, sebaiknya gunakan paket komponen AWS VSS terbaru untuk meningkatkan keandalan dan kinerja snapshot yang konsisten dengan aplikasi pada instans Windows EC2 Anda.

> **Note**
>
> Untuk menginstal atau memperbarui paket `AwsVssComponents` secara otomatis setiap kali Anda membuat snapshot yang konsisten dengan aplikasi, sebaiknya gunakan Systems Manager untuk menjalankan dokumen `AWSEC2-VssInstallAndSnapshot`. Untuk informasi selengkapnya, lihat Jalan AWSEC2-VssInstallAndSnapshot kan dokumen perintah (disarankan).

Untuk menginstal komponen VSS pada instans Windows Amazon EC2, ikuti langkah-langkah untuk lingkungan pilihan Anda.

Console

Untuk memasang komponen VSS menggunakan Distributor SSM

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih Jalankan Perintah.
3. Pilih Jalankan perintah.
4. Untuk Dokumen Perintah, pilih tombol di sebelah AWS-ConfigureAWSPackage.
5. Untuk Parameter perintah, lakukan hal berikut:
   a. Verifikasi bahwa Tindakan diatur menjadi Pasang.
   b. Untuk Nama, masukkan `AwsVssComponents`.
   c. Untuk Versi, masukkan versi atau kosongkan kolom sehingga System Manager menginstal versi terbaru.
6. Untuk Target, identifikasi instans di mana Anda ingin menjalankan operasi ini dengan menentukan tanda atau memilih instans secara manual.

   > **Note**
   >
   > Jika Anda memilih instans secara manual, dan instans yang ingin Anda lihat tidak disertakan dalam daftar, lihat Di Mana Instans Saya? dalam Panduan Pengguna AWS Systems Manager untuk kiat pemecahan masalah.

7. Untuk Parameter lainnya:
   - (Opsional) Untuk Komentar, ketik informasi tentang perintah ini.
   - Untuk Waktu habis (detik), tentukan jumlah detik untuk menunggu sistem sebelum gagal menjalankan perintah keseluruhan.
8. (Opsional) Untuk Kontrol laju:
   - Untuk Konkurensi, tentukan jumlah atau persentase instans untuk menjalankan perintah pada saat yang sama.

     > **Note**
     >
     > Jika Anda memilih target dengan memilih tanda Amazon EC2 dan Anda tidak yakin berapa banyak instans menggunakan tanda yang dipilih, batasi jumlah instans

yang dapat menjalankan dokumen pada waktu yang sama dengan menentukan persentase.

- Untuk Ambang kesalahan, tetapkan kapan harus berhenti menjalankan perintah pada instans lain setelah gagal pada jumlah atau persentase instans. Misalnya, jika Anda menentukan tiga kesalahan, Systems Manager berhenti mengirim perintah ketika kesalahan keempat diterima. Instans yang masih memproses perintah juga dapat mengirim kesalahan.

9. (Opsional) Untuk bagian Opsi output, jika Anda ingin menyimpan output perintah ke file, pilih kotak di samping Aktifkan penulisan ke bucket S3. Tentukan nama bucket dan nama prefiks (folder) (opsional).

> **Note**
>
> Izin S3 yang memberikan kemampuan untuk menulis data ke bucket S3 berasal dari profil instans yang ditetapkan ke instans, bukan data pengguna yang melaksanakan tugas ini. Untuk informasi selengkapnya, lihat Mengonfigurasi izin instans EC2 di Panduan AWS Systems Manager Pengguna.

10. (Opsional) Tentukan opsi untuk Notifikasi SNS.

    Untuk informasi tentang konfigurasi notifikasi Amazon SNS untuk Run Command, lihat Mengonfigurasi Notifikasi Amazon SNS untuk AWS Systems Manager.

11. Pilih Jalankan.

AWS CLI

Gunakan prosedur berikut untuk mengunduh dan menginstal paket `AwsVssComponents` pada instans Anda dengan menggunakan Run Command dari AWS CLI. Paket menginstal dua komponen: pemohon VSS dan penyedia VSS. Sistem menyalin komponen ini ke direktori pada instans, lalu mendaftarkan penyedia DLL sebagai penyedia VSS.

Untuk menginstal paket VSS

Jalankan perintah berikut untuk mengunduh dan menginstal komponen VSS yang diperlukan untuk System Manager.

```
aws ssm send-command \
```

```
    --document-name "AWS-ConfigureAWSPackage" \
    --instance-ids "i-1234567890abcdef0" \
    --parameters '{"action":["Install"],"name":["AwsVssComponents"]}'
```

PowerShell

Gunakan prosedur berikut untuk mengunduh dan menginstal `AwsVssComponents` paket pada instans Anda dengan menggunakan Run Command dari Tools untuk Windows PowerShell. Paket menginstal dua komponen: pemohon VSS dan penyedia VSS. Sistem menyalin komponen ini ke direktori pada instans, lalu mendaftarkan penyedia DLL sebagai penyedia VSS.

Untuk menginstal paket VSS

Jalankan perintah berikut untuk mengunduh dan menginstal komponen VSS yang diperlukan untuk System Manager.

```
Send-SSMCommand `
    -DocumentName "AWS-ConfigureAWSPackage" `
    -InstanceId "i-1234567890abcdef0" `
    -Parameter @{'action'='Install';'name'='AwsVssComponents'}
```

Verifikasi tanda tangan pada AWS Komponen VSS

Gunakan prosedur berikut untuk memverifikasi tanda tangan pada paket `AwsVssComponents`.

1. Hubungkan ke instans Windows Anda. Untuk informasi selengkapnya, lihat Hubungkan ke instans Windows Anda menggunakan RDP.
2. Arahkan ke C:\Program Files\ Amazon\AwsVssComponents.
3. Buka menu konteks (klik kanan) untuk `ec2-vss-agent.exe`, lalu pilih Properti.
4. Arahkan ke tab Tanda Tangan Digital dan verifikasi bahwa nama penandatangan adalah Amazon Web Services Inc.
5. Gunakan langkah-langkah sebelumnya untuk memverifikasi tanda tangan pada `Ec2VssInstaller` dan `Ec2VssProvider.dll`.

Memperbarui paket komponen VSS pada instans Windows EC2 Anda

Kami merekomendasikan Anda untuk terus meningkatkan komponen VSS dengan versi terbaru yang direkomendasikan. Ada beberapa cara berbeda untuk memperbarui komponen saat versi baru paket `AwsVssComponents` dirilis.

Metode pembaruan

- Anda dapat mengulangi langkah-langkah yang dijelaskan Instal komponen VSS secara manual pada instance EC2 Windows saat versi baru komponen AWS VSS dirilis.
- Anda dapat mengonfigurasi kaitan Systems Manager State Manager untuk secara otomatis mengunduh dan menginstal komponen VSS baru atau yang diperbarui saat paket `AwsVssComponents` tersedia.
- Anda dapat menginstal atau memperbarui paket `AwsVssComponents` secara otomatis setiap kali Anda membuat snapshot yang konsisten dengan aplikasi, sebaiknya gunakan Systems Manager untuk menjalankan dokumen `AWSEC2-VssInstallAndSnapshot`.

> **Note**
>
> Kami sarankan Anda menggunakan Systems Manager untuk menjalankan dokumen perintah `AWSEC2-VssInstallAndSnapshot`, yang secara otomatis menginstal atau memperbarui paket `AwsVssComponents` sebelum membuat snapshot yang konsisten dengan aplikasi. Untuk informasi selengkapnya, lihat Jalan AWSEC2-VssInstallAndSnapshot kan dokumen perintah (disarankan).

Untuk membuat kaitan Systems Manager State Manager, ikuti langkah-langkah untuk lingkungan pilihan Anda.

Console

Saat Anda membuat asosiasi Manajer Status Manajer Sistem, ada dua opsi untuk memperbarui `AwsVssComponents` paket, sebagai berikut:

Copot pemasangan dan instal ulang

Metode ini mengunduh dan menginstal paket tanpa prasyarat tambahan.

In-place memperbarui

Ini melakukan pembaruan di tempat untuk paket, dan memiliki prasyarat berikut:

- Versi SSM Agent yang diinstal pada instance harus versi `3.3.808.0` atau yang lebih baru. Untuk informasi selengkapnya, lihat Bekerja dengan SSM Agent di instans EC2 untuk Windows Server di Panduan Pengguna AWS Systems Manager .

- Jika ditentukan, versi `AwsVssComponents` paket harus versi `2.5.0` atau yang lebih baru. Versi sebelumnya tidak mendukung pembaruan di tempat.

> **Note**
>
> jika instans Anda tidak memenuhi prasyarat ini, pembaruan di tempat akan gagal. Gunakan opsi Uninstall dan instal ulang sebagai gantinya.

Untuk membuat asosiasi State Manager

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, pilih State Manager.

   Atau, jika beranda Systems Manager terbuka terlebih dahulu, buka panel navigasi lalu pilih State Manager.
3. Pilih Buat asosiasi.
4. Di bidang Nama, masukkan nama deskriptif.
5. Di daftar Dokumen, pilih AWS-ConfigureAWSPackage.
6. Di bagian Parameter, pilih Instal dari daftar Tindakan.
7. Untuk Jenis Instalasi, pilih Copot pemasangan dan instal ulang atau per In-place barui.
8. Di bidang Nama, masukkan `AwsVssComponents`. Anda dapat membuat bidang Versi dan Argumen Tambahan tetap kosong.
9. Di bagian Target, pilih opsi.

   > **Note**
   >
   > Jika Anda memilih untuk menargetkan instans dengan menggunakan tanda, dan Anda menentukan tanda yang memetakan ke instans Linux, kaitan berhasil pada instans Windows tetapi gagal pada instans Linux. Status keseluruhan asosiasi menunjukkan Gagal.
10. Di bagian Tentukan jadwal, pilih opsi.
11. Di bagian Opsi lanjutan, untuk Keparahan kepatuhan, pilih tingkat keparahan untuk kaitan. Untuk informasi selengkapnya, lihat Mem pelajari tentang kepatuhan asosiasi. Untuk Ubah

Kalender, pilih kalender dengan perubahan yang telah dikonfigurasi sebelumnya. Untuk informasi selengkapnya, lihat tentang Kalender Perubahan AWS Systems Manager.

12. Untuk kontrol Tarif, lakukan hal berikut:

    - Untuk Konkurensi, tentukan jumlah atau persentase dari simpul terkelola untuk menjalankan perintah pada saat yang sama.
    - Untuk Ambang kesalahan, tentukan kapan harus berhenti menjalankan perintah pada simpul terkelola lain setelah gagal pada jumlah atau persentase simpul.

13. (Opsional) Untuk Opsi output, untuk menyimpan output perintah ke file, pilih Aktifkan output penulisan ke S3. Masukkan nama bucket dan prefiks (folder) di dalam kotak.
14. Pilih Buat asosiasi, lalu pilih Tutup. Sistem ini mencoba untuk membuat asosiasi pada instans dan segera menerapkan status.

> **Note**
>
> Jika instans EC2 untuk Windows Server menunjukkan status Gagal, verifikasi bahwa Agen SSM berjalan pada instans, dan verifikasi bahwa instance dikonfigurasi dengan peran AWS Identity and Access Management (IAM) untuk Manajer Sistem. Untuk informasi selengkapnya, lihat Menyiapkan AWS Systems Manager.

AWS CLI

Gunakan perintah create-association untuk memperbarui paket Distributor sesuai jadwal tanpa membuat aplikasi terkait offline. Hanya file baru atau yang diperbarui dalam paket yang diganti.

Untuk membuat asosiasi State Manager

Jalankan perintah berikut untuk membuat asosiasi. Nilai `--name`, nama dokumen, selalu `AWS-ConfigureAWSPackage`. Perintah berikut menggunakan kunci `InstanceIds` untuk menentukan instans target.

```
aws ssm create-association \
    --name "AWS-ConfigureAWSPackage" \
    --parameters action=Install,installationType="Uninstall and
 reinstall",name=AwsVssComponents \
    --targets Key=InstanceIds,Values=i-1234567890abcdef0,i-000011112222abcde
```

PowerShell

Untuk membuat asosiasi State Manager

Gunakan New-SSMAssociation cmdlet.

```
New-SSMAssociation `
    -Name "AWS-ConfigureAWSPackage" `
    -Parameter  @{
        "action" = "Install"
        "installationType" = "Uninstall and reinstall"
        "name" = "AwsVssComponents"
    } `
    -Target @{
        "Key" = "InstanceIds"
        "Values" = @("i-1234567890abcdef0", "i-000011112222abcde")
    }
```

## Buat snapshot EBS berbasis VSS untuk instans Windows EC2 Anda

Setelah Anda memenuhi semuaPrasyarat untuk membuat snapshot EBS berbasis Windows VSS, Anda dapat menggunakan salah satu metode berikut untuk membuat snapshot berbasis VSS dari instans EC2 Anda.

AWS Systems Manager dokumen perintah

Gunakan dokumen perintah Manajer Sistemuntuk membuat snapshot berbasis VSS.

Untuk mengotomatiskan pencadangan, Anda dapat membuat tugas jendela AWS Systems Manager pemeliharaan yang menggunakan dokumen `AWSEC2-VssInstallAndSnapshot` perintah. Untuk informasi selengkapnya, lihat Bekerja dengan Jendela Pemeliharaan (Konsol) dalam Panduan Pengguna AWS Systems Manager .

AWS Backup

Anda dapat membuat cadangan VSS saat menggunakan AWS Backup dengan mengaktifkan VSS di konsol atau CLI. Untuk informasi selengkapnya, lihat Membuat cadangan VSS Windows di Panduan Developer AWS Backup .

> **Note**
>
> AWS Backup tidak secara otomatis menginstal `AwsVssComponents` paket pada instans Anda. Anda harus melakukan instalasi manual pada instans. Untuk informasi selengkapnya, lihat Instal komponen VSS secara manual pada instance EC2 Windows.

Amazon Data Lifecycle Manager

Anda dapat membuat snapshot VSS menggunakan Amazon Data Lifecycle Manager dengan mengaktifkan skrip pra dan pasca dalam kebijakan siklus hidup snapshot Anda. Untuk informasi selengkapnya, lihat Men gotomatiskan snapshot yang konsisten dengan aplikasi di Panduan Pengguna Amazon EBS.

> **Note**
>
> Amazon Data Lifecycle Manager tidak secara otomatis menginstal paket `AwsVssComponents` pada instans Anda. Anda harus melakukan instalasi manual pada instans. Untuk informasi selengkapnya, lihat Instal komponen VSS secara manual pada instance EC2 Windows.

## Gunakan dokumen perintah Manajer Sistem untuk membuat snapshot berbasis VSS

Anda dapat menggunakan dokumen AWS Systems Manager perintah untuk membuat snapshot berbasis VSS. Konten berikut memperkenalkan dokumen perintah yang tersedia, dan parameter runtime yang digunakan dokumen tersebut untuk membuat snapshot Anda.

Sebelum Anda menggunakan salah satu dokumen perintah Systems Manager, pastikan bahwa Anda telah memenuhi semua Prasyarat untuk membuat snapshot EBS berbasis Windows VSS.

Topik

- Parameter untuk dokumen snapshot VSS Systems Manager
- Jalankan dokumen perintah snapshot VSS Systems Manager

Parameter untuk dokumen snapshot VSS Systems Manager

Dokumen Systems Manager yang membuat snapshot VSS semuanya menggunakan parameter berikut, kecuali jika diberi catatan:

AmiName(string, opsional)

Jika CreateAmi opsi disetel ke`True`, tentukan nama AMI yang dibuat cadangan.

description (string, opsional)

Tentukan deskripsi untuk snapshot atau gambar yang dibuat proses ini.

CollectDiagnosticLogs(string, opsional)

Untuk mengumpulkan informasi lebih lanjut selama langkah pembuatan snapshot dan AMI, atur parameter ini ke "`True`". Nilai default untuk parameter ini adalah "`False`". Log diagnostik konsolidasi disimpan sebagai arsip `.zip` format di lokasi berikut pada instans Anda:

`C:\ProgramData\Amazon\AwsVss\Logs\`*`timestamp`*`.zip`

CopyOnly(string, opsional)

Jika Anda menggunakan cadangan SQL Server asli selain AWS VSS, melakukan pen Copy-only cadangan mencegah AWS VSS memutus rantai cadangan diferensial asli. Untuk melakukan operasi pen Copy-only cadangan, atur parameter ini ke`True`.

Nilai default untuk parameter ini adalah`False`, yang menyebabkan AWS VSS melakukan operasi pencadangan penuh.

CreateAmi(string, opsional)

Untuk membuat Amazon Machine Image (AMI) berbasis VSS untuk membuat cadangan instans Anda, setel parameter ini ke`True`. Nilai default untuk parameter ini adalah `False`, yang mencadangkan instans Anda dengan snapshot EBS sebagai gantinya.

Untuk informasi selengkapnya tentang cara membuat AMI dari suatu instans, lihat Buat EBS-backed AMI Amazon.

executionTimeout (string, opsional)

Tentukan waktu maksimum dalam hitungan detik untuk menjalankan proses pembuatan snapshot pada instans, atau untuk membuat AMI dari instans. Meningkatkan batas waktu ini memungkinkan perintah menunggu lebih lama hingga VSS memulai pembekuan dan melengkapi penandaan

sumber daya yang dibuatnya. Batas waktu ini hanya berlaku untuk langkah pembuatan snapshot atau AMI. Langkah awal untuk menginstal atau memperbarui paket `AwsVssComponents` tidak termasuk dalam batas waktu.

ExcludeBootVolume(string, opsional)

Pengaturan ini mengecualikan volume boot dari proses pencadangan jika Anda membuat snapshot. Untuk mengecualikan volume boot dari snapshot Anda, setel ExcludeBootVolume ke`True`, dan CreateAmi ke`False`.

Jika Anda membuat AMI untuk cadangan Anda, parameter ini harus diatur ke `False`. Nilai default untuk parameter ini adalah `False`.

NoWriters(string, opsional)

Untuk mengecualikan penulis VSS aplikasi dari proses snapshot, atur parameter ini ke `True`. Ini dapat membantu Anda mengatasi konflik dengan komponen cadangan VSS pihak ketiga. Nilai default untuk parameter ini adalah `False`.

Jika `SaveVssMetadata` ya`True`, parameter ini harus diatur ke`False`.

SaveVssMetadata(string, opsional)

Untuk menyimpan file metadata VSS selama setiap snapshot, atur parameter ini ke`True`. Nilai default-nya adalah `False`. File-file ini membantu memberikan wawasan tentang komponen atau penulis mana yang disertakan dalam operasi pencadangan, dan file serta volume terkait untuk setiap snapshot. File metadata digunakan saat memulihkan database SQL menggunakan solusi pemulihan VSS. Untuk informasi selengkapnya tentang cara memulihkan database SQL dari snapshot VSS, lihat Menggunakan runbook otomatisasi untuk memulihkan database Anda dari snapshot solusi AWS VSS.

File metadata memiliki id set snapshot terkait dalam namanya. Anda dapat menemukannya di lokasi berikut di instans Anda:

```
C:\ProgramData\Amazon\AwsVss\VssMetadata\
```

**Warning**

- Menyimpan file metadata VSS memerlukan `AwsVssComponents` paket versi 2.4.0 atau yang lebih baru. Jika instans Anda memiliki versi sebelumnya yang diinstal, pengaturan `SaveVssMetadata` untuk `True` menyebabkan pembuatan snapshot gagal.

- `SaveVssMetadata`Param `NoWriters` eter dan saling eksklusif. Jika keduanya disetel ke `True` maka pembuatan snapshot gagal.

tanda (string, opsional)

Sebaiknya beri tag snapshot dan gambar untuk membantu Anda menemukan dan mengelola sumber daya, misalnya, memulihkan volume dari daftar snapshot. Sistem menambahkan `Name` kunci, dengan nilai kosong di mana Anda dapat menentukan nama yang ingin Anda terapkan ke snapshot atau gambar keluaran Anda.

Jika Anda ingin menentukan tag tambahan, pisahkan tag dengan titik koma di antaranya. Misalnya, `Key=Environment,Value=Test;Key=User,Value=TestUser1`.

> **Note**
>
> Kunci dan nilai tag hanya boleh berisi karakter alfanumerik dan karakter khusus berikut:`( ) ./\-"'@_+:={}`.

Secara default, sistem menambahkan tag cadangan berikut untuk snapshot dan gambar berbasis VSS.

- Perangkat — Untuk snapshot berbasis VSS, ini adalah nama perangkat volume EBS yang diambil snapshot.
- AppConsistent— Tag ini menunjukkan keberhasilan pembuatan snapshot berbasis VSS atau AMI.
- AwsVssConfig— Ini mengidentifikasi snapshot dan AMI yang dibuat dengan VSS diaktifkan. Tag menyertakan informasi meta seperti `AwsVssComponents` versi, dan Snapshot Set ID.

> **Warning**
>
> Menentukan salah satu tag cadangan ini dalam daftar parameter Anda akan menyebabkan kesalahan.

VssVersion(string, opsional)

Khusus untuk dokumen `AWSEC2-VssInstallAndSnapshot`, Anda dapat menentukan parameter `VssVersion` guna menginstal versi paket `AwsVssComponents` tertentu pada instans. Biarkan parameter ini kosong untuk menginstal versi default yang direkomendasikan.

Jika versi paket `AwsVssComponents` yang ditentukan sudah diinstal, skrip melewatkan langkah penginstalan dan melanjutkan ke langkah pencadangan. Untuk daftar versi `AwsVssComponents` paket dan dukungan operasi, lihat AWS riwayat versi VSS solution.

Jalankan dokumen perintah snapshot VSS Systems Manager

Anda dapat membuat snapshot EBS berbasis VSS dengan dokumen AWS Systems Manager perintah sebagai berikut.

Jalan AWSEC2-VssInstallAndSnapshot kan dokumen perintah (disarankan)

Saat Anda menggunakan AWS Systems Manager untuk menjalankan `AWSEC2-VssInstallAndSnapshot` dokumen, skrip menjalankan langkah-langkah berikut.

1. Skrip terlebih dahulu menginstal atau memperbarui paket `AwsVssComponents` pada instans Anda, tergantung apakah sudah diinstal.
2. Skrip membuat snapshot yang konsisten dengan aplikasi setelah langkah pertama selesai.

Untuk menjalankan dokumen `AWSEC2-VssInstallAndSnapshot`, ikuti langkah-langkah untuk lingkungan pilihan Anda.

Console

Buat snapshot EBS berbasis VSS dari konsol

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Pilih Jalankan Perintah dari panel navigasi. Ini menunjukkan daftar perintah yang sedang berjalan di akun Anda, jika berlaku.
3. Pilih Jalankan perintah. Ini membuka daftar dokumen perintah yang dapat Anda akses.
4. Pilih `AWSEC2-VssInstallAndSnapshot` dari daftar dokumen perintah. Untuk merampingkan hasil, Anda dapat memasukkan semua atau sebagian dari nama dokumen. Anda juga dapat memfilter berdasarkan pemilik, berdasarkan jenis platform, atau dengan tanda.

Saat Anda memilih dokumen perintah, detail terisi di bawah daftar.

5. Pilih `Default version at runtime` dari daftar Versi dokumen.
6. Konfigurasikan Parameter perintah untuk menentukan cara `AWSEC2-VssInstallAndSnapshot` akan menginstal paket `AwsVssComponents` dan membuat cadangan dengan snapshot VSS atau AMI. Untuk detail parameter, lihat Parameter untuk dokumen snapshot VSS Systems Manager.
7. Untuk pemilihan target, tentukan tanda atau pilih instans secara manual untuk mengidentifikasi instans untuk menjalankan operasi ini.

   > **Note**
   >
   > Jika Anda memilih instans secara manual, dan instans yang ingin Anda lihat tidak disertakan dalam daftar, lihat Di Mana Instans Saya? untuk kiat pemecahan masalah.

8. Untuk parameter tambahan guna menentukan perilaku Run Command Systems Manager seperti Kontrol laju, masukkan nilai seperti yang dijelaskan dalam Menjalankan perintah dari konsol.
9. Pilih Jalankan.

   Jika berhasil, perintah tersebut akan mengisi daftar snapshot EBS dengan snapshot baru. Anda dapat menemukan snapshot ini di daftar snapshot EBS dengan mencari tanda yang Anda tentukan, atau dengan mencari `AppConsistent`. Jika pelaksanaan perintah gagal, lihat output perintah Systems Manager untuk detail tentang alasan pelaksanaan tersebut gagal. Jika perintah berhasil diselesaikan, tetapi pencadangan volume tertentu gagal, Anda dapat memecahkan masalah kegagalan tersebut dalam daftar volume EBS.

AWS CLI

Anda dapat menjalankan perintah berikut AWS CLI untuk membuat snapshot EBS berbasis VSS dan mendapatkan status pembuatan snapshot Anda.

Buat snapshot EBS berbasis VSS

Jalankan perintah berikut untuk membuat snapshot EBS berbasis VSS. Untuk membuat snapshot, Anda harus mengidentifikasi instans dengan parameter `--instance-ids`. Untuk informasi selengkapnya tentang parameter lain yang dapat Anda gunakan, lihat Parameter untuk dokumen snapshot VSS Systems Manager.

```
aws ssm send-command \
 --document-name "AWSEC2-VssInstallAndSnapshot" \
 --instance-ids "i-01234567890abcdef" \
 --parameters '{"ExcludeBootVolume":["False"],"description":["Description"],"tags":
["Key=key_name,Value=tag_value"],"VssVersion":[""]}'
```

Jika berhasil, dokumen perintah tersebut mengisi daftar snapshot EBS dengan snapshot baru. Anda dapat menemukan snapshot ini di daftar snapshot EBS dengan mencari tanda yang Anda tentukan, atau dengan mencari `AppConsistent`. Jika pelaksanaan perintah gagal, lihat output perintah untuk detail tentang alasan pelaksanaan tersebut gagal.

Dapatkan status perintah

Untuk mendapatkan status snapshot saat ini, jalankan perintah berikut menggunakan ID perintah yang dikembalikan dari send-command.

```
aws ssm get-command-invocation
 --instance-ids "i-01234567890abcdef" \
 --command-id "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" \
 --plugin-name "CreateVssSnapshot"
```

PowerShell

Jalankan perintah berikut AWS Tools for Windows PowerShell untuk membuat snapshot EBS berbasis VSS dan mendapatkan status runtime saat ini untuk pembuatan output Anda. Tentukan parameter yang dijelaskan dalam daftar sebelumnya untuk mengubah perilaku proses snapshot.

Buat snapshot EBS berbasis VSS dengan Tools untuk Windows PowerShell

Jalankan perintah berikut untuk membuat snapshot EBS berbasis VSS atau AMI.

```
Send-SSMCommand -DocumentName "AWSEC2-VssInstallAndSnapshot" -InstanceId
 "i-01234567890abcdef" -Parameter
 @{'ExcludeBootVolume'='False';'description'='a_description'
 ;'tags'='Key=key_name,Value=tag_value';'VssVersion'=''}
```

Dapatkan status perintah

Untuk mendapatkan status snapshot saat ini, jalankan perintah berikut menggunakan ID perintah yang dikembalikan dari Send-SSMCommand.

```
Get-SSMCommandInvocationDetail -InstanceId "i-01234567890abcdef" -CommandId
 "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111" -PluginName "CreateVssSnapshot"
```

Jika berhasil, perintah tersebut akan mengisi daftar snapshot EBS dengan snapshot baru. Anda dapat menemukan snapshot ini di daftar snapshot EBS dengan mencari tanda yang Anda tentukan, atau dengan mencari `AppConsistent`. Jika pelaksanaan perintah gagal, lihat output perintah untuk detail tentang alasan pelaksanaan tersebut gagal.

Jalankan dokumen AWSEC2-CreateVssSnapshot perintah

Untuk menjalankan dokumen `AWSEC2-CreateVssSnapshot`, ikuti langkah-langkah untuk lingkungan pilihan Anda.

Console

Buat snapshot EBS berbasis VSS dari konsol

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Pilih Jalankan Perintah dari panel navigasi. Ini menunjukkan daftar perintah yang sedang berjalan di akun Anda, jika berlaku.
3. Pilih Jalankan perintah. Ini membuka daftar dokumen perintah yang dapat Anda akses.
4. Pilih `AWSEC2-CreateVssSnapshot` dari daftar dokumen perintah. Untuk merampingkan hasil, Anda dapat memasukkan semua atau sebagian dari nama dokumen. Anda juga dapat memfilter berdasarkan pemilik, berdasarkan jenis platform, atau dengan tanda.

   Saat Anda memilih dokumen perintah, detail terisi di bawah daftar.
5. Pilih `Default version at runtime` dari daftar Versi dokumen.
6. Konfigurasikan Parameter perintah untuk menentukan cara `AWSEC2-CreateVssSnapshot` akan mencadangkan dengan snapshot VSS atau AMI. Untuk detail parameter, lihat Parameter untuk dokumen snapshot VSS Systems Manager.
7. Untuk pemilihan target, tentukan tanda atau pilih instans secara manual untuk mengidentifikasi instans untuk menjalankan operasi ini.

> **Note**
>
> Jika Anda memilih instans secara manual, dan instans yang ingin Anda lihat tidak disertakan dalam daftar, lihat Di Mana Instans Saya? untuk kiat pemecahan masalah.

8. Untuk parameter tambahan guna menentukan perilaku Run Command Systems Manager seperti Kontrol laju, masukkan nilai seperti yang dijelaskan dalam Menjalankan perintah dari konsol.
9. Pilih Jalankan.

   Jika berhasil, perintah tersebut akan mengisi daftar snapshot EBS dengan snapshot baru. Anda dapat menemukan snapshot ini di daftar snapshot EBS dengan mencari tanda yang Anda tentukan, atau dengan mencari `AppConsistent`. Jika pelaksanaan perintah gagal, lihat output perintah Systems Manager untuk detail tentang alasan pelaksanaan tersebut gagal. Jika perintah berhasil diselesaikan, tetapi pencadangan volume tertentu gagal, Anda dapat memecahkan masalah kegagalan tersebut dalam daftar volume EBS.

AWS CLI

Anda dapat menjalankan perintah berikut AWS CLI untuk membuat snapshot EBS berbasis VSS.

Buat snapshot EBS berbasis VSS

Jalankan perintah berikut untuk membuat snapshot EBS berbasis VSS. Untuk membuat snapshot, Anda harus mengidentifikasi instans dengan parameter `--instance-ids`. Untuk informasi selengkapnya tentang parameter lain yang dapat Anda gunakan, lihat Parameter untuk dokumen snapshot VSS Systems Manager.

```
aws ssm send-command \
 --document-name "AWSEC2-CreateVssSnapshot" \
 --instance-ids "i-01234567890abcdef" \
 --parameters '{"ExcludeBootVolume":["False"],"description":["Description"],"tags":
["Key=key_name,Value=tag_value"]}'
```

Jika berhasil, dokumen perintah tersebut mengisi daftar snapshot EBS dengan snapshot baru. Anda dapat menemukan snapshot ini di daftar snapshot EBS dengan mencari tanda yang Anda tentukan, atau dengan mencari `AppConsistent`. Jika pelaksanaan perintah gagal, lihat output perintah untuk detail tentang alasan pelaksanaan tersebut gagal.

PowerShell

Jalankan perintah berikut dengan AWS Tools for Windows PowerShell untuk membuat snapshot EBS berbasis VSS.

Buat snapshot EBS berbasis VSS dengan Tools untuk Windows PowerShell

Jalankan perintah berikut untuk membuat snapshot EBS berbasis VSS. Untuk membuat snapshot, Anda harus mengidentifikasi instans dengan parameter `InstanceId`. Anda dapat menentukan lebih dari satu instans untuk membuat snapshot. Untuk informasi selengkapnya tentang parameter lain yang dapat Anda gunakan, lihat Parameter untuk dokumen snapshot VSS Systems Manager.

```
Send-SSMCommand -DocumentName AWSEC2-CreateVssSnapshot -InstanceId
 "i-01234567890abcdef" -Parameter
 @{'ExcludeBootVolume'='False';'description'='a_description'
 ;'tags'='Key=key_name,Value=tag_value'}
```

Jika berhasil, perintah tersebut akan mengisi daftar snapshot EBS dengan snapshot baru. Anda dapat menemukan snapshot ini di daftar snapshot EBS dengan mencari tanda yang Anda tentukan, atau dengan mencari `AppConsistent`. Jika pelaksanaan perintah gagal, lihat output perintah untuk detail tentang alasan pelaksanaan tersebut gagal. Jika perintah berhasil diselesaikan, tetapi pencadangan volume tertentu gagal, Anda dapat memecahkan masalah kegagalan dalam daftar snapshot EBS.

Jalankan dokumen perintah untuk Klaster Failover Windows dengan penyimpanan EBS bersama

Anda dapat menggunakan salah satu prosedur baris perintah yang dijelaskan di bagian sebelumnya untuk membuat snapshot berbasis VSS. Dokumen perintah (`AWSEC2-VssInstallAndSnapshot` atau `AWSEC2-CreateVssSnapshot`) harus berjalan pada simpul primer di klaster Anda. Dokumen akan gagal pada simpul sekunder karena tidak memiliki akses ke disk bersama. Jika primer dan sekunder berubah secara dinamis, Anda dapat menjalankan dokumen AWS Systems Manager Run Command pada beberapa node dengan harapan bahwa perintah akan berhasil pada node primer dan gagal pada node sekunder.

> **Note**
>
> Untuk mengotomatiskan pencadangan, Anda dapat membuat tugas jendela AWS Systems Manager pemeliharaan yang menggunakan `AWSEC2-VssInstallAndSnapshot` dokumen.

Untuk informasi selengkapnya, lihat Bekerja dengan Jendela Pemeliharaan (Konsol) dalam Panduan Pengguna AWS Systems Manager .

## Memecahkan masalah snapshot EBS berbasis Windows VSS

Sebelum Anda mencoba langkah pemecahan masalah lainnya, sebaiknya Anda memverifikasi detail berikut.

- Pastikan bahwa Anda telah memenuhi semua Prasyarat untuk membuat snapshot EBS berbasis Windows VSS.
- Verifikasi bahwa Anda menggunakan Dukungan versi Windows OS paket `AwsVssComponents` terbaru untuk sistem operasi Anda. Masalah yang Anda lihat mungkin telah diatasi di versi yang lebih baru.

Topik


- Periksa file log
- Kumpulkan log diagnostik tambahan
- Gunakan VSS pada instans dengan proxy yang dikonfigurasi
- Kesalahan: Koneksi pipa thaw kehabisan waktu, kesalahan pada thaw, batas waktu menunggu VSS Freeze, atau kesalahan batas waktu lainnya
- Kesalahan: Tidak dapat menginvokasi metode. Invokasi metode hanya didukung pada tipe inti dalam mode bahasa ini


### Periksa file log

Jika Anda mengalami masalah atau menerima pesan kesalahan saat membuat snapshot EBS berbasis VSS, Anda dapat melihat output perintah di konsol Manajer Sistem.

Untuk dokumen Manajer Sistem yang membuat snapshot VSS, Anda dapat mengatur `CollectDiagnosticLogs` parameter ke "`True`" saat runtime. Ketika `CollectDiagnosticLogs` parameter disetel ke "`True`“, VSS mengumpulkan log tambahan untuk membantu dalam debugging. Untuk informasi selengkapnya, lihat Kumpulkan log diagnostik tambahan.

Jika Anda mengumpulkan log diagnostik, dokumen Manajer Sistem menyimpannya di instance Anda di lokasi berikut:`C:\ProgramData\Amazon\AwsVss\Logs\`*`timestamp`*`.zip`. Default untuk `CollectDiagnosticLogs` parameter adalah "`False`".

> **Note**
>
> Untuk bantuan debugging tambahan, Anda dapat mengirim `.zip` file ke Dukungan.

Log tambahan berikut tersedia, apakah Anda mengumpulkan log diagnostik atau tidak:

- `%ProgramData%\Amazon\SSM\InstanceData\`*`InstanceID`*`\document\orchestration\`*`SSMCommandID`*`\awsrunPowerShellScript\runPowerShellScript\stdout`
- `%ProgramData%\Amazon\SSM\InstanceData\`*`InstanceID`*`\document\orchestration\`*`SSMCommandID`*`\awsrunPowerShellScript\runPowerShellScript\stderr`

Anda juga dapat membuka aplikasi Event Viewer Windows dan memilih Log Windows, Aplikasi untuk melihat log tambahan. Untuk melihat peristiwa khusus dari Penyedia VSS Windows EC2 dan Layanan Salin Bayangan Volume, filter berdasarkan Sumber pada **`Ec2VssSoftwareProvider`** dan **`VSS`**.

Jika Anda menggunakan System Manager dengan titik akhir VPC, dan tindakan API send-command System Manager (J alankan Perintah di konsol) gagal, verifikasi bahwa Anda mengonfigurasi titik akhir berikut dengan benar: com.amazonaws. ***`region`***.ec2.

Tanpa titik akhir Amazon EC2 yang ditentukan, panggilan untuk menghitung volume EBS terlampir gagal, yang menyebabkan perintah Manajer Sistem gagal. Untuk informasi selengkapnya tentang pengaturan titik akhir VPC dengan Systems Manager, lihat Buat Titik Akhir Virtual Private Cloud dalam Panduan Pengguna AWS Systems Manager .

## Kumpulkan log diagnostik tambahan

Untuk mengumpulkan log diagnostik tambahan saat Anda menggunakan perintah kirim Manajer Sistem untuk menjalankan dokumen snapshot VSS, atur parameter `CollectDiagnosticLogs` input ke "`True`" saat runtime. Kami menyarankan Anda mengatur parameter ini ke "`True`" saat Anda memecahkan masalah.

Untuk melihat contoh baris perintah, pilih salah satu tab berikut.

AWS CLI

Contoh berikut menjalankan dokumen Manajer `AWSEC2-CreateVssSnapshot` Sistem di AWS CLI:

```
aws ssm send-command \
    --document-name "AWSEC2-CreateVssSnapshot" \
    --instance-ids "i-1234567890abcdef0" \
    --parameters '{"description":["Example - create diagnostic logs at
 runtime."],"tags":["Key=tag_name,Value=tag_value"],"CollectDiagnosticLogs":
["True"]}'
```

PowerShell

Contoh berikut menjalankan dokumen Manajer `AWSEC2-CreateVssSnapshot` Sistem di PowerShell:

```
Send-SSMCommand `
    -DocumentName "AWSEC2-CreateVssSnapshot" `
    -InstanceId "i-1234567890abcdef0" `
    -Parameter @{'description'='Example - create diagnostic logs at
 runtime.';'tags'='Key=tag_name,Value=tag_value';'CollectDiagnosticLogs'='True'}
```

## Gunakan VSS pada instans dengan proxy yang dikonfigurasi

Jika Anda mengalami masalah saat membuat snapshot EBS berbasis VSS pada instans yang menggunakan proxy untuk mencapai titik akhir EC2, verifikasi pengaturan berikut pada instans Anda:

- Verifikasi bahwa proxy dikonfigurasi sehingga titik akhir layanan EC2 di Region dan IMDS instans dapat dijangkau dengan AWS Tools for Windows PowerShell menjalankan sebagai SYSTEM.
- Untuk mendukung penggunaan proxy WinHTTP yang dikonfigurasi sistem, pastikan Anda telah menginstal `AwsVssComponents` versi terbaru pada instans Anda. Untuk informasi selengkapnya tentang mengonfigurasi proksi WinHTTP, lihat Netsh Commands for Windows Hypertext Transfer Protocol (WINHTTP) di situs web Microsoft.

## Kesalahan: Koneksi pipa thaw kehabisan waktu, kesalahan pada thaw, batas waktu menunggu VSS Freeze, atau kesalahan batas waktu lainnya

Penyedia VSS Windows EC2 mungkin habis karena aktivitas atau layanan pada instans yang mencegah snapshot berbasis VSS berjalan tepat waktu. Windows VSS Framework menyediakan jendela 10 detik yang tidak dapat dikonfigurasi selama komunikasi ke sistem file dijeda. Selama waktu ini, `AWSEC2-CreateVssSnapshot` snapshot volume Anda.

Masalah-masalah berikut dapat menyebabkan EC2 Windows VSS Provider mengalami waktu habis selama snapshot:

- Ber I/O lebihan hingga volume
- Responsif lambat dari API EC2 pada instans
- Volume terfragmentasi
- Ketidakcocokan dengan beberapa perangkat lunak antivirus
- Masalah dengan penulis aplikasi VSS
- Ketika Module Logging diaktifkan untuk sejumlah besar PowerShell modul, itu dapat menyebabkan PowerShell skrip berjalan lambat

Sebagian besar masalah waktu habis yang terjadi saat Anda menjalankan dokumen perintah `AWSEC2-CreateVssSnapshot` berkaitan dengan beban kerja pada instans yang terlalu tinggi pada saat pencadangan. Tindakan berikut dapat membantu Anda mengambil snapshot dengan sukses:

- Coba lagi perintah `AWSEC2-CreateVssSnapshot` untuk melihat apakah upaya snapshot berhasil. Jika mencoba kembali berhasil dalam beberapa kasus, mengurangi beban instans mungkin membuat snapshot lebih berhasil.
- Tunggu beberapa saat untuk mendapatkan penurunan beban kerja pada instans, dan coba lagi perintah `AWSEC2-CreateVssSnapshot`. Atau, Anda dapat mencoba snapshot ketika instans diketahui berada di bawah tekanan rendah.
- Mencoba snapshot VSS saat perangkat lunak antivirus pada sistem dimatikan. Jika ini menyelesaikan masalah, lihat petunjuk perangkat lunak antivirus dan konfigurasikan untuk memungkinkan Snapshot VSS.
- Jika ada panggilan API Amazon EC2 bervolume tinggi di akun Anda dalam Wilayah yang sama tempat Anda menjalankan snapshot, throttling API mungkin menunda operasi snapshot. Untuk mengurangi dampak pelambatan, gunakan `AwsVssComponents` paket terbaru. Paket ini

menggunakan tindakan `CreateSnapshots` API EC2 untuk mengurangi jumlah tindakan mutasi seperti pembuatan snapshot per volume dan penandaan.

- Jika Anda memiliki lebih dari satu skrip perintah `AWSEC2-CreateVssSnapshot` yang berjalan secara bersamaan, Anda dapat mengambil langkah berikut untuk mengurangi masalah konkurensi.
  - Pertimbangkan untuk menjadwalkan snapshot selama periode aktivitas API yang lebih rendah.
  - Jika Anda menggunakan Run Command di konsol Systems Manager (atau SendCommand di API) untuk menjalankan skrip perintah, Anda dapat menggunakan kontrol laju Systems Manager guna mengurangi konkurensi.

    Anda juga dapat menggunakan kontrol tarif Manajer Sistem untuk mengurangi konkurensi untuk layanan seperti AWS Backup itu menggunakan Manajer Sistem untuk menjalankan skrip perintah.
- Jalankan perintah `vssadmin list writers` dalam shell dan lihat apakah laporan kesalahan dalam kolom Kesalahan terakhir lapangan untuk setiap penulis pada sistem. Jika ada penulis melaporkan waktu habis, pertimbangkan untuk mencoba kembali snapshot ketika beban instans sedang rendah.
- Bila Anda menggunakan jenis instans yang lebih kecil seper *`t2 | t3 | t3a`* ti.nano at *`t2 | t3 | t3a`* au.micro, batas waktu karena kendala memori dan CPU dapat terjadi. Tindakan berikut dapat membantu mengurangi masalah waktu habis.
  - Coba tutup aplikasi intensif memori atau CPU sebelum mengambil snapshot.
  - Coba ambil snapshot selama periode aktivitas instans yang lebih rendah.

## Kesalahan: Tidak dapat menginvokasi metode. Invokasi metode hanya didukung pada tipe inti dalam mode bahasa ini

Anda akan mengalami kesalahan ini ketika mode PowerShell bahasa tidak diatur ke`FullLanguage`. Dok `AWSEC2-CreateVssSnapshot` umen SSM harus PowerShell dikonfigurasi ke `FullLanguage` mode.

Untuk memverifikasi mode bahasa, jalankan perintah berikut pada instance di PowerShell konsol:

```
$ExecutionContext.SessionState.LanguageMode
```

Untuk informasi selengkapnya tentang mode bahasa, lihat about_ Language_Modes dalam dokumentasi Microsoft.

# Gunakan AWS Solusi VSS untuk memulihkan data untuk instance Anda

Anda dapat memulihkan volume EBS untuk instance Windows dari snapshot berbasis VSS yang dibuat oleh solusi VSS. AWS Jika snapshot solusi AWS VSS Anda berisi cadangan database Microsoft SQL Server, Anda dapat memulihkan database menggunakan runbook `AWSEC2-RestoreSqlServerDatabaseWithVss` AWS Systems Manager Otomasi.

Runbook pemulihan database mengotomatiskan seluruh proses pemulihan, termasuk membuat volume dari snapshot dan melampirkannya ke instance. Otomatisasi memanfaatkan teknologi VSS untuk memulihkan database, memungkinkan Anda memulihkan tanpa menghentikan aplikasi SQL Server Anda atau memutuskan koneksi aktif apa pun.

Untuk petunjuk terperinci tentang cara menggunakan runbook database Microsoft SQL Server, lihat Memuli hkan dari snapshot berbasis VSS di Panduan Pengguna Microsoft SQL Server di Amazon EC2.

## Menyesuaikan skrip untuk memulihkan volume EBS dari AWS Snapshot solusi VSS

Anda dapat menggunakan `RestoreVssSnapshotSampleScript.ps1` skrip sebagai model untuk membuat skrip kustom Anda sendiri yang mengembalikan volume EBS dari snapshot solusi AWS VSS. Skrip sampel melakukan tugas-tugas berikut:

- Menghentikan suatu instans
- Hapus semua drive yang ada dari instans (kecuali volume boot, jika dikecualikan)
- Membuat volume baru dari snapshot
- Melampirkan volume ke instans dengan menggunakan tanda ID perangkat di snapshot
- Memulai ulang instans.

> **Important**
>
> Skrip berikut ini memisahkan semua volume yang terlampir ke suatu instans, lalu membuat volume baru dari snapshot. Pastikan bahwa Anda telah mencadangkan instans' dengan benar. Volume lama tidak dihapus. Jika mau, Anda dapat mengedit skrip untuk menghapus volume lama.

Untuk memulihkan volume dari snapshot EBS berbasis VSS

1. Unduh RestoreVssSnapshotSampleScript.zip berkas dan mengekstrak konten berkas.
2. Buka `RestoreVssSnapshotSampleScript.ps1` di editor teks dan edit panggilan sampel di bagian bawah skrip dengan ID instans EC2 dan ID snapshot EBS yang valid, kemudian jalankan skrip dari PowerShell.

## AWS riwayat versi VSS solution

Halaman ini mencakup catatan rilis berdasarkan versi untuk paket komponen AWS VSS, serta persyaratan versi komponen dan skrip untuk setiap versi Windows Server yang didukung.

Topik

- AwsVssComponents versi paket
- Dukungan versi Windows OS

### AwsVssComponents versi paket

Tabel berikut menjelaskan versi rilis dari paket komponen AWS VSS.

| Versi | Detail | Tanggal rilis | Dapat diunduh |
|---|---|---|---|
| 2.5.2 | • Menambahkan dukungan untuk titik akhir `aws-eusc` partisi.<br>• AWS SDK yang diperbarui untuk memasukkan pembaruan keamanan. | 2 September 2026 | Ya |
| 2.5.1 | Memperbaiki kasus di mana pemulihan database SQL dapat gagal ketika parameter database target ditentukan. | Maret 13, 2025 | Ya |
| 2.5.0 | • | Januari 17, 2025 | Ya |

| Versi | Detail | Tanggal rilis | Dapat diunduh |
|---|---|---|---|
| | Menambahkan kemampuan untuk membaca file metadata VSS dan mengembalikan database Microsoft SQL Server pada instance. Untuk informasi selengkapnya, lihat Memulihkan dari snapshot berbasis VSS di Panduan Pengguna Microsoft SQL Server di Amazon EC2.<br>• Menambahkan dukungan untuk opsi pembaruan di tempat saat menginstal atau memutakhirkan `AwsVssComponents` paket. | | |
| 2.4.0 | Menambahkan kemampuan untuk menyimpan file metadata VSS pada pembuatan snapshot. Untuk mengaktifkan fitur ini, lihat SaveVssMetadata diParameter untuk dokumen snapshot VSS Systems Manager. | Oktober 7, 2024 | Ya |
| 2.3.3 | Memperbarui agen VSS untuk memastikan bahwa `Ec2VssProvider` digunakan selama pembuatan snapshot. | Juni 25, 2024 | Ya |
| 2.3.2 | Memperbaiki kasus di mana pendaftaran penyedia VSS tidak dihapus pada penghapusan instalasi. | 9 Mei 2024 | Ya |
| 2.3.1 | Menambahkan tag default baru `AwsVssConfig` untuk mengidentifikasi snapshot dan AMI yang dibuat oleh AWS VSS. | Maret 7, 2024 | Ya |

| Versi | Detail | Tanggal rilis | Dapat diunduh |
| --- | --- | --- | --- |
| 2.2.1 | • Menambahkan dukungan untuk menggunakan `DescribeInstanceAttribute` API.<br>• Perbaikan bug dan peningkatan keandalan.<br>• Dukungan tidak digunakan lagi untuk Windows Server 2012 dan 2012 R2. AWS Instalasi komponen VSS versi 2.2.1 pada Windows Server 2012 dan 2012 R2 akan gagal. AWS Komponen VSS versi 2.1.0 adalah versi terakhir untuk mendukung Windows Server 2012 dan 2012 R2. | Januari 18, 2024 | Ya |
| 2.1.0 | Menambahkan dukungan untuk menggunakan `CreateSnapshots` API. | 6 November 2023 | Ya |
| 2.0.1 | Dukungan tambahan untuk menggunakan pengaturan proksi WinHTTP. | 26 Oktober 2023 | Tidak |
| 2.0.0 | Menambahkan kemampuan ke komponen AWS VSS untuk membuat snapshot dan AMI, yang memungkin kan kompatibilitas dengan logging PowerShell modul, logging blok skrip, dan fitur transkripsi. | 28 April 2023 | Tidak |
| 1.3.2.0 | Memperbaiki kasus di mana kegagalan instalasi tidak dilaporkan dengan benar. | 10 Mei 2022 | Tidak |

| Versi | Detail | Tanggal rilis | Dapat diunduh |
| --- | --- | --- | --- |
| 1.3.1.0 | • Snapshot tetap gagal di pengendali domain dalam kaitannya dengan kesalahan logging penulis NTDS VSS.<br>• Kesalahan agen VSS tetap saat menghapus pemasangan penyedia VSS versi 1.0. | 6 Februari 2020 | Ya |
| 1.3.00 | • Penebangan yang lebih baik dengan mengurangi kata benda yang tidak diinginkan.<br>• Masalah wilayah diperbaiki selama instalasi.<br>• Memperbaiki kode pengembalian untuk beberapa kondisi kesalahan pendaftaran penyedia.<br>• Memperbaiki berbagai masalah instalasi. | 19 Maret 2019 | Tidak |
| 1.2.00 | • Menambahkan parameter baris perintah `-nw` (tidak-menulis) dan `-copy` (hanya-salinan) kepada agen.<br>• Memperbaiki EventLog kesalahan yang disebabkan oleh panggilan alokasi memori yang tidak tepat. | 15 November 2018 | Tidak |
| 1.1 | Memperbaiki komponen AWS VSS yang digunakan secara tidak benar sebagai penyedia Cadangan dan Pemulihan Windows default. | 12 Desember 2017 | Tidak |
| 1.0 | Rilis awal. | 20 November 2017 | Tidak |

## Dukungan versi Windows OS

Tabel berikut menunjukkan versi solusi AWS VSS mana yang harus Anda jalankan pada setiap versi Windows Server di Amazon EC2.

| Versi Windows Server | AwsVssComponents versi | AWSEC2-VssInstallAndSnapshot nama versi | AWSEC2-CreateVssSnapshot nama versi |
|---|---|---|---|
| Windows Server 2025 | default | default | default |
| Windows Server 2022 | default | default | default |
| Windows Server 2019 | default | default | default |
| Windows Server 2016 | default | default | default |
| Windows Server 2012 R2 | 2.1.0 | Tidak didukung | 2012R2 |
| Windows Server 2012 | 2.1.0 | Tidak didukung | 2012R2 |
| Windows Server 2008 R2 | 1.3.1.0 | Tidak didukung | 2008R2 |

# Penyimpanan objek, penyimpanan file, dan caching file di Amazon EC2

Penyimpanan file cloud adalah metode untuk menyimpan data di cloud yang menyediakan akses server dan aplikasi ke data melalui sistem file bersama. Kompatibilitas ini membuat penyimpanan file cloud ideal untuk beban kerja yang mengandalkan sistem file bersama dan menyediakan integrasi sederhana tanpa perubahan kode.

Ada banyak solusi penyimpanan file yang ada, mulai dari server file node tunggal pada instance komputasi menggunakan penyimpanan blok sebagai dasar tanpa skalabilitas atau sedikit redundansi untuk melindungi data, hingga solusi do-it-yourself berkerumun, hingga solusi yang dikelola sepenuhnya. Konten berikut memperkenalkan beberapa layanan penyimpanan yang disediakan oleh AWS untuk digunakan dengan EC2 instans Amazon.

Konten


- Menggunakan Amazon S3 dengan instans Amazon EC2
- Menggunakan Amazon EFS dengan instans Amazon EC2 Linux
- Gunakan Amazon FSx dengan instans Amazon EC2
- Menggunakan Cache File Amazon dengan EC2 instans Amazon


## Menggunakan Amazon S3 dengan instans Amazon EC2

Amazon Simple Storage Service (Amazon S3) adalah layanan penyimpanan objek yang menawarkan skalabilitas, ketersediaan data, keamanan, dan kinerja terdepan di industri. Anda dapat menggunakan Amazon S3 untuk menyimpan dan mengambil sejumlah data untuk berbagai kasus penggunaan, seperti data lake, situs web, cadangan, dan analisis data besar, dari instans Amazon EC2 atau dari mana saja melalui internet. Untuk informasi selengkapnya, lihat Apa itu Amazon S3?

Ada dua cara untuk mengakses data Amazon S3 dari instans Amazon EC2 Anda:

- Akses file — Gunakan Amazon S3 Files untuk memasang bucket S3 sebagai sistem file berkinerja tinggi pada instans Anda.
- Akses objek — Gunakan Amazon S3 API, AWS CLI AWS SDK, atau alat seperti wget untuk menyalin objek ke dan dari S3.

## Akses file dengan Amazon S3 Files

Amazon S3 Files adalah sistem file tanpa server yang memungkinkan Anda memasang bucket tujuan umum S3 Anda sebagai sistem file berkinerja tinggi pada instans komputasi Anda. Dengan File S3, Anda dapat mengakses objek S3 Anda sebagai file dengan menggunakan operasi sistem file standar seperti membaca dan menulis di jalur pemasangan lokal.

Anda dapat memasang sistem file S3 ke instans EC2 baik saat peluncuran, atau setelah peluncuran pada instance yang sedang berjalan.

Prasyarat

Sebelum Anda mengatur File S3 dengan instans EC2 Anda, pastikan Anda memiliki yang berikut:

- Sistem file S3 dan setidaknya satu target mount dalam status yang tersedia. Untuk informasi tentang membuat sistem file S3, lihat Bek erja dengan File Amazon S3 di Panduan Pengguna Amazon S3.
- Instans Linux EC2 dengan profil instance yang dilampirkan padanya. Untuk informasi tentang izin yang diperlukan untuk memasang sistem file, lihat peran dan kebijakan IAM di Panduan Pengguna Amazon S3.
- Grup keamanan yang mengizinkan lalu lintas NFS (port 2049) antara instance Anda dan target mount sistem file. Untuk informasi tentang pengaturan grup keamanan yang diperlukan, lihat Grup keamanan di Panduan Pengguna Amazon S3.

Untuk memasang sistem file ke instance EC2 saat peluncuran menggunakan konsol EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Luncurkan instans.
3. Pada Pengaturan jaringan, lakukan hal berikut:

    a. Pilih Edit.

    b. Untuk Subnet, pilih subnet.

    c. Pilih grup keamanan default untuk memastikan bahwa instance EC2 Anda dapat mengakses sistem file S3 Anda. Anda tidak dapat mengakses instans EC2 dengan Secure Shell (SSH) menggunakan grup keamanan ini. Untuk akses dengan SSH, Anda nantinya dapat mengedit grup keamanan default dan menambahkan aturan untuk mengizinkan SSH, atau menambahkan grup keamanan baru yang memungkinkan SSH. Anda dapat menggunakan pengaturan berikut:

i. Jenis: SSH

ii. Protokol: TCP

iii. Rentang Pelabuhan: 22

iv. Sumber: Any where 0.0.0. 0/0

4. Di bawah Konfigurasi penyimpanan, lakukan hal berikut:

   a. Di bawah Sistem file, pilih File S3.

   b. Pilih Tambahkan sistem file bersama.

   c. Untuk sistem file S3, sistem file Anda muncul di Availability Zone berdasarkan subnet yang Anda pilih di setelan Jaringan. Pilih sistem file S3 yang ingin Anda pasang. Jika Anda tidak memiliki sistem file apa pun, pilih Buat sistem file baru untuk membuat yang baru.

   d. Masukkan jalur pemasangan lokal pada instans EC2 tempat Anda ingin memasang sistem file (misalnya,`/mnt/s3files`).

   e. Sebuah perintah akan dihasilkan untuk memasang sistem file dan menambahkannya ke fstab. Anda dapat menambahkan perintah ini ke bidang Data pengguna di bawah Detail lanjutan. Instans EC2 Anda kemudian akan dikonfigurasi untuk memasang sistem file S3 saat diluncurkan dan setiap kali di-reboot. Anda juga dapat menjalankan perintah ini di instance EC2 Anda setelah diluncurkan.

5. Di bawah Detail lanj utan, lampirkan profil instance ke instance Anda. Peran IAM Anda harus memiliki izin untuk memasang sistem file dan mengakses bucket S3. Untuk informasi selengkapnya tentang izin yang diperlukan, lihat peran dan kebijakan IAM di Panduan Pengguna Amazon S3.

6. Pilih Luncurkan instans.

   Setelah instance diluncurkan, utilitas perangkat lunak yang diperlukan diinstal dan sistem file dipasang. Anda dapat melihat sistem file dengan menavigasi ke jalur pemasangan lokal Anda.

Untuk memasang sistem file ke instance EC2 setelah peluncuran

1. Sambungkan ke instans EC2 Anda melalui Secure Shell (SSH) atau dengan menggunakan EC2 Instance Connect di konsol EC2.

2. Untuk memasang sistem file S3 Anda, gunakan utilitas mount helper`amazon-efs-utils`. Tergantung pada distribusi Linux Anda, gunakan salah satu perintah berikut untuk menginstal `amazon-efs-utils` paket:

a. Jika Anda menggunakan Amazon Linux, jalankan perintah berikut untuk menginstal efs-utils dari repositori Amazon:

```
sudo yum -y install amazon-efs-utils
```

b. Jika Anda menggunakan distribusi Linux lain yang didukung, jalankan perintah berikut:

```
curl https://amazon-efs-utils.aws.com/efs-utils-installer.sh | sudo sh -s -- --
install
```

c. Untuk distribusi Linux lainnya, lihat repositori efs-utils pada GitHub.

3. Buat direktori untuk titik pemasangan sistem file menggunakan perintah berikut:

```
sudo mkdir {path/to/mount}
```

4. Pasang sistem file S3:

```
FS="{YOUR_FILE_SYSTEM_ID}"
sudo mount -t s3files $FS:/ {path/to/mount}
```

5. Konfirmasikan sistem file sudah terpasang:

```
df -h {path/to/mount}
```

Untuk melihat objek di bucket S3 Anda sebagai file

Setelah menyelesaikan prosedur sebelumnya, Anda sekarang dapat membaca dan menulis objek S3 sebagai file di jalur pemasangan lokal Anda menggunakan operasi sistem file standar. Jika Anda memiliki objek di bucket S3 Anda, Anda dapat melihatnya sebagai file dengan menggunakan perintah berikut:

```
ls {path/to/mount}
```

## Object-based akses

Anda dapat menyalin file ke dan dari Amazon S3 menggunakan API S3 AWS CLI, AWS SDK, atau alat HTTP standar. Jika Anda memiliki izin yang diperlukan, Anda dapat menyalin file ke atau dari Amazon S3 dan instans Anda menggunakan salah satu metode berikut.

wget

> **Note**
>
> Metode ini hanya berfungsi untuk objek publik. Jika objek tidak publik, Anda menerima pesan `ERROR 403: Forbidden`. Jika Anda menerima kesalahan ini, Anda harus menggunakan konsol Amazon S3, AWS API AWS CLI, AWS SDK, atau AWS Tools for Windows PowerShell, dan Anda harus memiliki izin yang diperlukan. Untuk informasi selengkapnya, lihat Identitas dan manajemen akses untuk Amazon S3 dan Mengunduh objek di Panduan Pengguna Amazon S3.

wgetUtilitas ini adalah klien HTTP dan FTP yang dapat Anda gunakan untuk mengunduh objek publik dari Amazon S3. Utilitas ini diinstal secara default di Amazon Linux dan sebagian besar distribusi lainnya, dan dapat diunduh di Windows. Untuk mengunduh objek Amazon S3, gunakan perintah berikut, dengan mengganti URL objek yang akan diunduh.

```
[ec2-user ~]$ wget https://amzn-s3-demo-bucket.s3.amazonaws.com/path-to-file
```

PowerShell

Anda dapat menggunakan AWS Tools for Windows PowerShell untuk memindahkan objek ke dan dari Amazon S3.

Gunakan Copy-S3Object cmdlet untuk menyalin objek Amazon S3 ke instance Windows Anda sebagai berikut.

```
Copy-S3Object `
    -BucketName amzn-s3-demo-bucket `
    -Key path-to-file `
    -LocalFile my_copied_file.ext
```

Atau, Anda dapat membuka konsol Amazon S3 dengan menggunakan browser web pada instance Windows.

### AWS CLI

Anda dapat menggunakan AWS Command Line Interface (AWS CLI) untuk mengunduh item terbatas dari Amazon S3 dan mengunggah item. Untuk informasi selengkapnya, seperti cara menginstal dan mengonfigurasi alat, lihat halaman detail AWS Command Line Interface.

Per intah aws s3 cp mirip dengan cp perintah Unix. Anda dapat menyalin file dari Amazon S3 ke instans Anda, menyalin file dari instans ke Amazon S3, dan menyalin file dari satu lokasi Amazon S3 ke lokasi lainnya.

Gunakan perintah berikut untuk menyalin objek dari Amazon S3 ke instance Anda:

```
aws s3 cp s3://amzn-s3-demo-bucket/my_folder/my_file.ext my_copied_file.ext
```

Gunakan perintah berikut untuk menyalin objek dari instance Anda kembali ke Amazon S3:

```
aws s3 cp my_copied_file.ext s3://amzn-s3-demo-bucket/my_folder/my_file.ext
```

Per intah aws s3 sync dapat menyinkronkan seluruh bucket Amazon S3 ke lokasi direktori lokal. Hal ini dapat membantu untuk mengunduh kumpulan data dan menjaga agar salinan lokal tetap mutakhir dengan set jarak jauh. Jika Anda memiliki izin yang tepat pada bucket Amazon S3, Anda dapat mendorong direktori lokal Anda kembali ke cloud setelah selesai dengan membalikkan lokasi sumber dan tujuan dalam perintah.

Gunakan perintah berikut untuk mengunduh seluruh bucket Amazon S3 ke direktori lokal pada instance Anda:

```
aws s3 sync s3://amzn-s3-demo-source-bucket local_directory
```

### Amazon S3 API

Anda dapat menggunakan API untuk mengakses data di Amazon S3. Anda dapat menggunakan API ini untuk membantu mengembangkan aplikasi Anda dan mengintegrasikannya dengan API dan SDK lain. Untuk informasi selengkapnya, lihat Contoh kode untuk Amazon S3 menggunakan AWS SDK di Refer ensi API Layanan Penyimpanan Sederhana Amazon.

# Menggunakan Amazon EFS dengan instans Amazon EC2 Linux

> **Note**
>
> Amazon EFS tidak didukung pada instans Windows.

Amazon EFS menyediakan penyimpanan file yang dapat diskalakan untuk digunakan bersama Amazon EC2. Anda dapat menggunakan sistem file EFS sebagai sumber data umum untuk beban kerja dan aplikasi yang berjalan pada beberapa instans. Untuk informasi selengkapnya, silakan lihat halaman produk Amazon Elastic File System.

Tutorial ini menunjukkan cara membuat dan melampirkan sistem file Amazon EFS menggunakan wizard Buat Cepat Amazon EFS selama peluncuran instance. Untuk tutorial tentang cara membuat sistem file menggunakan konsol Amazon EFS, lihat Memulai Amazon Elastic File System di Panduan Pengguna Amazon Elastic File System.

> **Note**
>
> Ketika Anda membuat sistem file EFS menggunakan Pembuatan Cepat EFS, sistem file dibuat dengan pengaturan layanan yang direkomendasikan berikut ini:
>
> - Pencadangan otomatis diaktifkan.
> - Kelola target pemasangan di VPC yang dipilih.
> - Mode kinerja Tujuan Umum.
> - Mode throughput yang meledak.
> - Enkripsi data saat istirahat diaktifkan menggunakan kunci default Anda untuk Amazon EFS (`aws/elasticfilesystem`).
> - Manajemen siklus hidup Amazon EFS diaktifkan dengan kebijakan 30 hari.

Tugas

- Membuat sistem file EFS menggunakan Amazon EFS Quick Create
- Menguji sistem file EFS
- Menghapus sistem file EFS

## Membuat sistem file EFS menggunakan Amazon EFS Quick Create

Anda dapat membuat sistem file EFS dan memasangnya ke instans saat meluncurkan instans Anda menggunakan fitur Amazon EFS Quick Create pada wizard peluncuran instans Amazon EC2.

Untuk membuat sistem file EFS menggunakan Amazon EFS Quick Create

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Luncurkan instans.
3. (Opsional) Di bawah Nama dan tanda, untuk Nama, masukkan nama untuk mengidentifikasi instans Anda.
4. Di bawah Aplikasi dan Gambar OS (Amazon Machine Image), pilih sistem operasi Linux, lalu untuk Amazon Machine Image (AMI), pilih AMI Linux.
5. Di bawah Tipe Instans, untuk Tipe Instans, pilih tipe instans atau pertahankan default.
6. Di bawah Pasangan kunci (login), untuk Nama pasangan kunci, pilih pasangan kunci yang sudah ada atau buat yang baru.
7. Di bawah Pengaturan jaringan, pilih Edit (di kanan), lalu untuk Subnet, pilih subnet.

   > **Note**
   >
   > Anda harus memilih subnet sebelum dapat menambahkan sistem file EFS.

8. Di bawah Konfigurasikan penyimpanan, pilih Edit (di kanan bawah), lalu lakukan hal berikut:

   a. Untuk Sistem file, pastikan EFS dipilih, lalu pilih Buat sistem file bersama baru.

   b. Untuk Nama sistem file masukkan nama untuk sistem file Amazon EFS, lalu pilih Buat sistem file.

   c. Untuk Mount point, tentukan titik pemasangan khusus atau pertahankan defaultnya.

   d. Untuk mengaktifkan akses ke sistem file, pilih Secara otomatis membuat dan melampirkan grup keamanan. Dengan memilih kotak centang ini, grup keamanan berikut akan secara otomatis dibuat dan dilampirkan ke instance dan target pemasangan sistem file:

      - Grup keamanan instans — Menyertakan aturan keluar yang mengizinkan lalu lintas melalui NFS 2049port, tetapi tidak menyertakan aturan masuk.
      - Grup keamanan target mount sistem file – Termasuk aturan masuk yang mengizinkan lalu lintas melalui port NFS 2049 dari grup keamanan instans (dijelaskan di atas), dan aturan keluar yang mengizinkan lalu lintas melalui port NFS 2049.

> **Note**
>
> Atau, Anda dapat membuat dan melampirkan grup keamanan secara manual. Jika Anda ingin membuat dan memasang grup keamanan yang Secara otomatis buat dan lampirkan grup keamanan yang diperlukan.

e. Untuk secara otomatis memasang sistem file bersama saat instans diluncurkan, pilih Pasang sistem file bersama secara otomatis dengan melampirkan skrip data pengguna yang diperlukan. Untuk melihat data pengguna yang dibuat secara otomatis, perluas Detail lanjutan, dan gulir ke bawah ke Data pengguna.

> **Note**
>
> Jika Anda menambahkan data pengguna sebelum memilih kotak centang ini, data pengguna asli akan diganti oleh data pengguna yang dibuat secara otomatis.

9. Konfigurasikan pengaturan instans lain sesuai kebutuhan.
10. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

## Menguji sistem file EFS

Anda dapat terhubung ke instance Anda dan memverifikasi bahwa sistem file dipasang ke direktori yang Anda tentukan (misalnya,/mnt/efs).

Untuk memverifikasi bahwa sistem file terpasang

1. Connect ke instans Anda. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..
2. Dari jendela terminal untuk instans, jalankan perintah df -T untuk memverifikasi bahwa sistem file EFS sudah terpasang.

```
$ df -T
Filesystem     Type             1K-blocks    Used          Available Use% Mounted
 on
/dev/xvda1     ext4               8123812 1949800            6073764  25% /
devtmpfs       devtmpfs           4078468      56            4078412   1% /dev
```

```
tmpfs           tmpfs                4089312        0             4089312   0% /dev/shm
efs-dns         nfs4       9007199254740992        0  9007199254740992   0% /mnt/efs
```

Perhatikan bahwa nama sistem file, yang ditunjukkan dalam contoh output sebagai*efs-dns*, memiliki bentuk berikut.

```
file-system-id.efs.aws-region.amazonaws.com:/
```

3. (Opsional) Buat file di sistem file dari instans, lalu verifikasi bahwa Anda dapat melihat file dari instans lain.

   a. Dari instans, jalankan perintah berikut untuk membuat file.

   ```
   $ sudo touch /mnt/efs/test-file.txt
   ```

   b. Dari instans lain, jalankan perintah berikut untuk melihat file.

   ```
   $ ls /mnt/efs
   test-file.txt
   ```

### Menghapus sistem file EFS

Jika Anda tidak lagi memerlukan sistem file, Anda dapat menghapusnya.

Untuk menghapus sistem file

1. Buka konsol Amazon Elastic File System di https://console.aws.amazon.com/efs/.
2. Pilih sistem file yang akan dihapus.
3. Pilih Tindakan, Hapus sistem file.
4. Saat diminta konfirmasi, masukkan ID sistem file dan pilih Hapus sistem file.

## Gunakan Amazon FSx dengan instans Amazon EC2

Rangkaian layanan Amazon FSx memudahkan peluncuran, pengoperasian, dan skala penyimpanan bersama yang didukung oleh sistem file komersial dan sumber terbuka yang populer. Anda dapat menggunakan wizard peluncuran instans baru untuk secara otomatis melampirkan jenis sistem file Amazon FSx berikut ke instans Amazon EC2 Anda saat peluncuran:

- Amazon FSx untuk NetApp ONTAP menyediakan penyimpanan bersama yang dikelola sepenuhnya di AWS Cloud dengan akses data populer dan kemampuan manajemen ONTAP. NetApp
- Amazon FSx for OpenZFS menyediakan penyimpanan bersama hemat biaya yang dikelola sepenuhnya dan ditenagai oleh sistem file OpenZFS yang populer.

> **Note**
>
> - Fungsi ini tersedia di wizard peluncuran instans baru saja. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol
> - Amazon FSx for Windows File Server dan sistem file Amazon FSx for Lustre tidak dapat dipasang saat peluncuran. Anda harus memasang sistem file ini secara manual setelah peluncuran.

Anda dapat memilih untuk memasang sistem file yang sudah ada yang Anda buat sebelumnya, atau Anda dapat membuat sistem file baru untuk dipasang ke instans saat peluncuran.

Topik

- Grup keamanan dan skrip data pengguna
- Memasang sistem file Amazon FSx saat peluncuran

## Grup keamanan dan skrip data pengguna

Saat Anda memasang sistem file Amazon FSx ke sebuah instans menggunakan wizard peluncuran instans, Anda dapat memilih apakah akan secara otomatis membuat dan melampirkan grup keamanan yang diperlukan untuk mengaktifkan akses ke sistem file, dan apakah akan secara otomatis menyertakan skrip data pengguna yang diperlukan untuk memasang sistem file dan membuatnya tersedia untuk digunakan.

Topik

- Grup keamanan
- Skrip data pengguna

Grup keamanan

Jika Anda memilih untuk secara otomatis membuat grup keamanan yang diperlukan untuk mengaktifkan akses ke sistem file, wizard peluncuran instans membuat dan melampirkan dua grup keamanan - satu grup keamanan dilampirkan ke instans, dan yang lainnya dilampirkan ke sistem file. Untuk informasi selengkapnya tentang persyaratan grup keamanan, lihat kontrol akses sistem file FSx untuk ONTAP dengan Amazon VPC dan FSx untuk kontrol akses sistem file OpenZFS dengan Amazon VPC.

Kami menambahkan tanda `Name=instance-sg-1` ke grup keamanan yang dibuat dan dilampirkan ke instans. Nilai dalam tanda secara otomatis dinaikkan setiap kali wizard peluncuran instans membuat grup keamanan untuk sistem file Amazon FSx.

Grup keamanan mencakup aturan output berikut ini, tetapi tidak ada aturan masuk.

Aturan-aturan ke luar

| Tipe protokol | Nomor port | Tujuan |
|---|---|---|
| UDP | 111 | *file system security group* |
| UDP | 20001 - 20003 | *file system security group* |
| UDP | 4049 | *file system security group* |
| UDP | 2049 | *file system security group* |
| UDP | 635 | *file system security group* |
| UDP | 4045 - 4046 | *file system security group* |
| TCP | 4049 | *file system security group* |
| TCP | 635 | *file system security group* |
| TCP | 2049 | *file system security group* |
| TCP | 111 | *file system security group* |
| TCP | 4045 - 4046 | *file system security group* |
| TCP | 20001 - 20003 | *file system security group* |

| Tipe protokol | Nomor port | Tujuan |
| --- | --- | --- |
| Semua | Semua | *file system security group* |

Grup keamanan yang dibuat dan dilampirkan ke sistem file ditandai dengan `Name=fsx-sg-`*1*. Nilai dalam tanda secara otomatis dinaikkan setiap kali wizard peluncuran instans membuat grup keamanan untuk sistem file Amazon FSx.

Grup keamanan mencakup aturan berikut.

Aturan-aturan ke dalam

| Tipe protokol | Nomor port | Sumber |
| --- | --- | --- |
| UDP | 2049 | *instance security group* |
| UDP | 20001 - 20003 | *instance security group* |
| UDP | 4049 | *instance security group* |
| UDP | 111 | *instance security group* |
| UDP | 635 | *instance security group* |
| UDP | 4045 - 4046 | *instance security group* |
| TCP | 4045 - 4046 | *instance security group* |
| TCP | 635 | *instance security group* |
| TCP | 2049 | *instance security group* |
| TCP | 4049 | *instance security group* |
| TCP | 20001 - 20003 | *instance security group* |
| TCP | 111 | *instance security group* |

Aturan-aturan ke luar

| Tipe protokol | Nomor port | Tujuan |
| --- | --- | --- |
| Semua | Semua | 0.0.0. 0/0 |

Skrip data pengguna

Jika Anda memilih untuk secara otomatis melampirkan skrip data pengguna, wizard peluncuran instans menambahkan data pengguna berikut ke instans. Skrip ini menginstal paket-paket yang diperlukan, memasang sistem file, dan memperbarui pengaturan instans Anda sehingga sistem file akan secara otomatis dipasang ulang setiap kali instans dimulai ulang.

```
#cloud-config
package_update: true
package_upgrade: true
runcmd:
- yum install -y nfs-utils
- apt-get -y install nfs-common
- svm_id_1=svm_id
- file_system_id_1=file_system_id
- vol_path_1=/vol1
- fsx_mount_point_1=/mnt/fsx/fs1
- mkdir -p "${fsx_mount_point_1}"
- if [ -z "$svm_id_1" ]; then printf "\n${file_system_id_1}.fsx.eu-
north-1.amazonaws.com:/${vol_path_1} ${fsx_mount_point_1} nfs4
 nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,_netdev
 0 0\n" >> /etc/fstab; else printf "\n${svm_id_1}.${file_system_id_1}.fsx.eu-
north-1.amazonaws.com:/${vol_path_1} ${fsx_mount_point_1} nfs4
 nfsvers=4.1,rsize=1048576,wsize=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0
 0\n" >> /etc/fstab; fi
- retryCnt=15; waitTime=30; while true; do mount -a -t nfs4 defaults; if [ $? = 0 ] ||
 [ $retryCnt -lt 1 ]; then echo File system mounted successfully; break; fi; echo File
 system not available, retrying to mount.; ((retryCnt--)); sleep $waitTime; done;
```

## Memasang sistem file Amazon FSx saat peluncuran

Untuk memasang sistem file Amazon FSx baru atau yang sudah ada saat peluncuran

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.

2. Di panel navigasi, pilih Instans dan kemudian pilih Luncurkan instans untuk membuka wizard peluncuran instans.
3. Di bagian Citra Aplikasi dan OS, pilih AMI yang akan digunakan.
4. Di bagian Tipe instans, pilih tipe instans.
5. Di bagian Pasangan kunci, pilih pasangan kunci yang sudah ada atau buat yang baru.
6. Di bagian Pengaturan jaringan, lakukan hal berikut ini:

   a. Pilih Edit.

   b. Jika Anda ingin memasang sistem file yang ada, untuk Subnet, pilih subnet pilihan sistem file. Sebaiknya luncurkan instans ke Zona Ketersediaan yang sama dengan subnet pilihan sistem file untuk mengoptimalkan performa.

      Jika Anda ingin membuat sistem file baru untuk dipasang ke sebuah instans, untuk Subnet, pilih subnet yang akan digunakan untuk meluncurkan instans.

      > **Important**
      >
      > Anda harus memilih subnet untuk mengaktifkan fungsionalitas Amazon FSx di wizard peluncuran instans yang baru. Jika Anda tidak memilih subnet, Anda tidak akan dapat memasang sistem file yang ada atau membuat yang baru.

7. Di bagian Penyimpanan, lakukan hal berikut ini:

   a. Konfigurasikan volume sesuai kebutuhan.

   b. Perluas bagian Sistem file dan pilih FSx.

   c. Pilih Tambahkan sistem file bersama.

   d. Untuk Sistem File, pilih sistem file yang akan dipasang.

      > **Note**
      >
      > Daftar ini menampilkan semua sistem file Amazon FSx untuk NetApp ONTAP dan Amazon FSx untuk OpenZFS di akun Anda di Wilayah yang dipilih.

   e. Untuk secara otomatis membuat dan melampirkan grup keamanan yang diperlukan untuk mengaktifkan akses ke sistem file, pilih Buat dan lampirkan grup keamanan secara otomatis. Jika Anda lebih suka membuat grup keamanan secara manual, kosongkan kotak centang. Untuk informasi selengkapnya, lihat Grup keamanan.

f. Untuk secara otomatis melampirkan skrip data pengguna yang diperlukan untuk memasang sistem file, pilih Secara otomatis memasang sistem file bersama dengan melampirkan skrip data pengguna yang diperlukan. Jika Anda lebih suka memberikan skrip data pengguna secara manual, kosongkan kotak centang. Untuk informasi selengkapnya, lihat Skrip data pengguna.

8. Di bagian Lanjutan, konfigurasikan pengaturan instans tambahan sesuai kebutuhan.
9. Pilih Luncurkan.

## Menggunakan Cache File Amazon dengan EC2 instans Amazon

Amazon File Cache menyediakan cache berkecepatan tinggi yang dikelola sepenuhnya AWS yang membuatnya lebih mudah untuk memproses data file, di mana pun data disimpan. Amazon File Cache berfungsi sebagai lokasi penyimpanan sementara berkinerja tinggi untuk data yang disimpan di sistem file lokal, sistem AWS file, dan bucket Amazon Simple Storage Service (Amazon S3). Anda dapat menggunakan kemampuan ini untuk membuat kumpulan data yang tersebar tersedia untuk aplikasi berbasis file AWS dengan tampilan terpadu, dan pada kecepatan tinggi—latensi sub-milidetik dan throughput tinggi. Untuk informasi selengkapnya, lihat Panduan Pengguna Cache File Amazon.

Amazon File Cache berfungsi dengan Linux paling populer AMIs, dan kompatibel dengan tipe instans berbasis x86 dan tipe instans Graviton. Anda dapat mengakses cache dari EC2 instans Amazon menggunakan klien Lustre open-source. Anda dapat me-mount cache Anda dan kemudian bekerja dengan file dan direktori di cache Anda menggunakan perintah Linux standar. EC2 Instans Amazon dapat mengakses cache Anda dari Availability Zone lain dalam virtual private cloud (VPC) yang sama, asalkan konfigurasi jaringan Anda memungkinkan akses di seluruh subnet dalam VPC. Anda juga dapat membuat cache di VPC bersama.

Untuk memulai, lihat Memulai Cache File Amazon di Panduan Pengguna Cache File Amazon.

# Kelola sumber daya Amazon EC2 Anda

Sumber daya adalah entitas yang dapat Anda kerjakan. Amazon EC2 membuat sumber daya saat Anda menggunakan fitur layanan. Misalnya, sumber daya Amazon EC2 mencakup gambar, instans, armada, pasangan kunci, dan grup keamanan. Semua jenis sumber daya Amazon EC2 menyertakan atribut yang menggambarkan sumber daya. Misalnya, nama, deskripsi, pengidentifikasi sumber daya, dan Nama Sumber Daya Amazon (ARN).

Sumber daya Amazon EC2 khusus untuk Wil AWS ayah atau zona tempat mereka berada. Misalnya, Amazon Machine Image (AMI) khusus untuk Wil AWS ayah, tetapi instance yang Anda luncurkan dari AMI khusus untuk zona di mana Anda meluncurkannya. Anda dapat menentukan sumber daya Amazon EC2 dalam kebijakan izin menggunakan ARN-nya.

Anda Akun AWS memiliki kuota default untuk Amazon EC2. Kuota ini menentukan jumlah maksimum sumber daya yang dapat Anda buat. Misalnya, ada kuota untuk jumlah maksimum vCPU di seluruh instans yang sedang berjalan. Jika meluncurkan instance atau memulai instans yang dihentikan akan menyebabkan Anda melebihi kuota, operasi gagal.

Anda dapat mencari sumber daya tertentu di Akun AWS menurut Wilayah, menggunakan ID sumber daya atau tag. Untuk mencari sumber daya atau jenis sumber daya tertentu di beberapa Wilayah, gunakan Tampilan Global Amazon EC2.

Daftar Isi


- Pilih Wilayah untuk sumber daya Amazon EC2 Anda
- Temukan sumber daya Amazon EC2 Anda
- Melihat sumber daya di seluruh Wilayah menggunakan AWS Tampilan Global
- Tandai sumber daya Amazon EC2 Anda
- Kuota layanan Amazon EC2


## Pilih Wilayah untuk sumber daya Amazon EC2 Anda

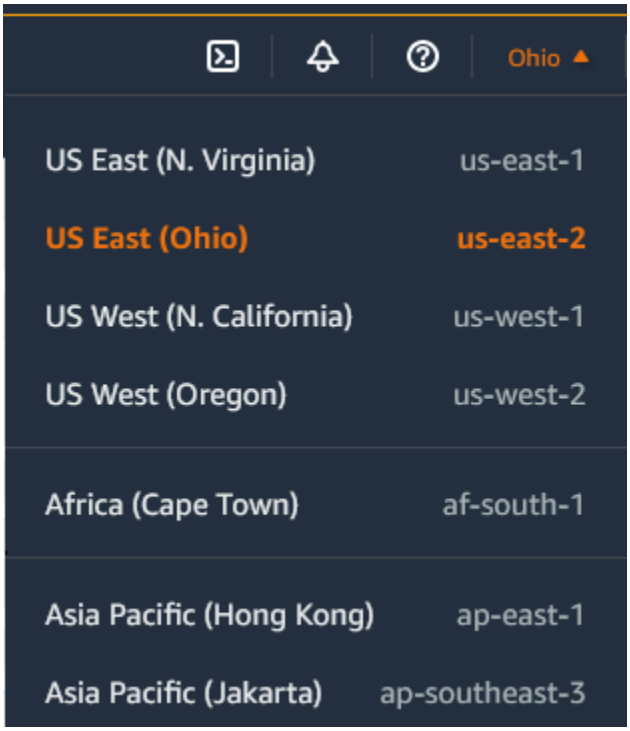
Sumber daya Amazon EC2 khusus untuk Wil AWS ayah atau zona tempat mereka berada. Saat Anda membuat sumber daya Amazon EC2, Anda memilih Wilayah untuk sumber daya.

Pertimbangan-pertimbangan

- Beberapa AWS sumber daya mungkin tidak tersedia di semua Wilayah. Pastikan Anda dapat membuat semua AWS sumber daya yang Anda butuhkan di Wilayah sebelum Anda mulai membuat sumber daya di Wilayah.
- Beberapa Wilayah dinonaktifkan secara default. Anda harus mengaktifkan Wilayah ini sebelum dapat menggunakannya. Untuk informasi selengkapnya, lihat AWS Wilayah.

Untuk memilih Wilayah untuk sumber daya menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi, pilih pemilih Wilayah, kemudian pilih Wilayah.




3. Pemilih Wilayah mencakup semua sumber daya yang tersedia untuk digunakan di Anda Akun AWS. Pilih teks yang digarisbawahi di dekat bagian bawah daftar untuk melihat Wilayah yang tidak diaktifkan untuk akun Anda.

Untuk memilih Wilayah untuk sumber daya menggunakan AWS CLI

Anda dapat mengonfigurasi AWS CLI untuk menggunakan Wilayah default. Jika Anda tidak menentukan Wilayah dalam perintah, Wilayah default digunakan. Untuk menggunakan Wilayah yang berbeda untuk perintah tertentu, tambahkan opsi berikut.

```
--region us-east-1
```

Untuk memilih Wilayah untuk sumber daya menggunakan Alat untuk PowerShell

Anda dapat mengonfigurasi Alat untuk Windows PowerShell untuk menggunakan Wilayah default. Jika Anda tidak menentukan Wilayah dalam perintah, Wilayah default digunakan. Untuk menggunakan Wilayah yang berbeda untuk perintah tertentu, tambahkan parameter berikut.

```
-Region us-east-1
```

# Temukan sumber daya Amazon EC2 Anda

Anda dapat memperoleh daftar beberapa tipe sumber daya menggunakan konsol Amazon EC2. Anda dapat memperoleh daftar setiap tipe sumber daya menggunakan perintah atau tindakan API yang sesuai. Jika memiliki banyak sumber daya, Anda dapat memfilter hasilnya agar hanya menyertakan atau mengecualikan sumber daya yang cocok dengan kriteria tertentu.

Daftar Isi


- Membuat daftar dan memfilter sumber daya menggunakan konsol
- Daftar dan filter menggunakan baris perintah
- Melihat sumber daya di seluruh Wilayah menggunakan AWS Tampilan Global


## Membuat daftar dan memfilter sumber daya menggunakan konsol

Daftar Isi

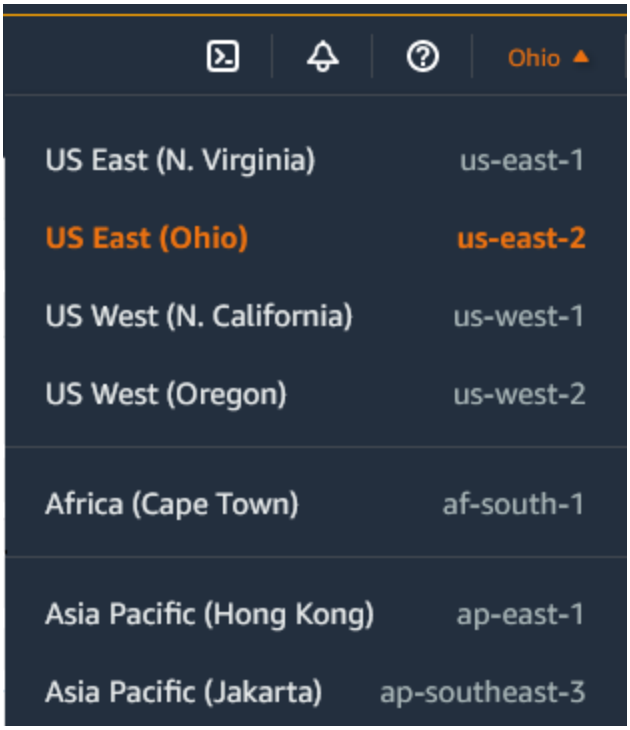

- Membuat daftar sumber daya menggunakan konsol
- Memfilter sumber daya menggunakan konsol
  - Filter yang didukung
- Simpan set filter menggunakan konsol
  - Fitur utama
  - Buat set filter
  - Memodifikasi set filter
  - Menghapus kumpulan filter

## Membuat daftar sumber daya menggunakan konsol

Anda dapat melihat tipe sumber daya Amazon EC2 yang paling umum menggunakan konsol. Untuk melihat sumber daya tambahan, gunakan antarmuka baris perintah atau tindakan API.

Untuk membuat daftar sumber daya EC2 menggunakan konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Sumber daya Amazon EC2 khusus untuk. Wilayah AWS Dari bilah navigasi, pilih Wilayah dari pem ilih Wilayah.



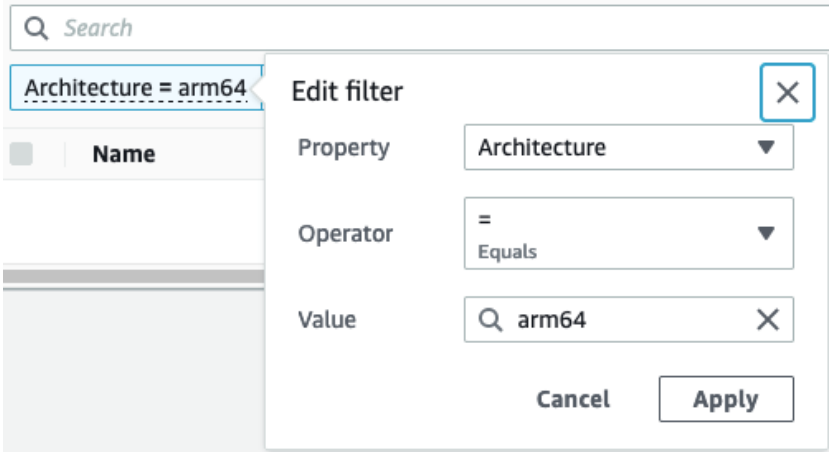

3. Di panel navigasi, pilih opsi yang sesuai dengan tipe sumber daya tersebut. Misalnya, untuk mencantumkan semua instance Anda, pilih Instans.

## Memfilter sumber daya menggunakan konsol

Untuk memfilter daftar sumber daya

1. Di panel navigasi, pilih tipe sumber daya (misalnya, Instans).
2. Pilih bidang pencarian.
3. Pilih filter dari dalam daftar.

4. Pilih operator, misalnya, = (Sama dengan). Beberapa atribut memiliki berbagai operator yang tersedia untuk dipilih. Perhatikan bahwa tidak semua layar mendukung pemilihan operator.
5. Pilih nilai filter.
6. Untuk mengedit filter yang dipilih, pilih token filter (kotak biru), lakukan pengeditan yang diperlukan, lalu pilih Terapkan. Perhatikan bahwa tidak semua layar mendukung pengeditan filter yang dipilih.




7. Setelah selesai, hapus filter.

Filter yang didukung

Konsol Amazon EC2 mendukung dua jenis penyaringan.

- Pemfilteran API terjadi pada sisi server. Pemfilteran diterapkan pada panggilan API, mengurangi jumlah sumber daya yang dikembalikan oleh server. Tindakan ini memungkinkan pemfilteran cepat di sejumlah set sumber daya dan tindakan ini akan dapat menghemat waktu transfer data serta biaya antara server dan peramban. Pemfilteran API mendukung operator = (sama dengan) dan : (berisi) dan selalu peka huruf besar/kecil.
- Pemfilteran klien terjadi pada sisi klien. Tindakan ini memungkinkan Anda untuk memfilter data yang sudah tersedia dalam peramban (dengan kata lain, data yang telah dikembalikan oleh API). Pemfilteran klien berfungsi baik dalam kaitannya dengan filter API untuk memfilter set data yang lebih kecil dalam peramban. Selain operator = (sama dengan) dan: (berisi), pemfilteran klien juga dapat mendukung berbagai operator, seperti >= (lebih besar dari atau sama dengan), dan operator negasi (terbalik), seperti!= (tidak sama dengan).

Konsol Amazon EC2 mendukung jenis pencarian berikut:

### Pencarian berdasarkan kata kunci

Pencarian berdasarkan kata kunci adalah pencarian teks bebas yang memungkinkan Anda mencari nilai di semua atribut atau tanda sumber daya, tanpa menentukan atribut atau kunci tanda untuk pencarian.

> **Note**
>
> Semua pencarian kata kunci menggunakan pemfilteran klien.

Untuk mencari berdasarkan kata kunci, masukkan atau tempelkan apa yang Anda cari dalam bidang pencarian, lalu pilih Enter. Misalnya, pencarian `123` cocok dengan semua instans yang memiliki 123 dalam atributnya, seperti alamat IP, ID instans, ID VPC, atau ID AMI, atau dalam tandanya, seperti Nama. Jika pencarian teks bebas Anda menampilkan kecocokan yang tidak terduga, terapkan filter tambahan.

### Cari berdasarkan atribut

Pencarian berdasarkan atribut memungkinkan Anda untuk mencari atribut tertentu pada semua sumber daya.

> **Note**
>
> Pencarian atribut menggunakan pemfilteran API atau pemfilteran klien, bergantung pada atribut yang dipilih. Saat melakukan pencarian atribut, atribut akan dikelompokkan.

Misalnya, Anda dapat mencari atribut Status Instans untuk semua instans agar hanya menampilkan instans yang berada dalam status `stopped`. Untuk melakukannya:

1. Di bidang pencarian pada layar Instans, mulai masukkan `Instance state`. Saat Anda memasukkan karakter, kedua tipe filter muncul untuk Status Instans: Filter API dan Filter Klien.
2. Untuk mencari pada sisi server, pilih Status instans di bawah Filter API. Untuk mencari pada sisi klien, pilih Status instans (klien) di bawah Filter klien.

   Daftar operator yang mungkin untuk atribut yang dipilih akan muncul.
3. Pilih operator = (Sama dengan).

   Daftar dari nilai yang mungkin untuk atribut dan operator yang dipilih akan muncul.

4. Pilih dihentikan dari daftar.

Cari berdasarkan tanda

Pencarian berdasarkan tanda memungkinkan Anda memfilter sumber daya dalam tabel yang ditampilkan saat ini berdasarkan kunci tanda atau nilai tanda.

Pencarian tanda menggunakan pemfilteran API atau pemfilteran klien, tergantung pengaturan di jendela Preferensi.

Guna memastikan pemfilteran API untuk tanda

1. Buka jendela Preferensi.
2. Kosongkan kotak centang Gunakan pencocokan ekspresi reguler. Jika kotak centang ini dipilih, pemfilteran klien dilakukan.
3. Pilih kotak centang Gunakan pencocokan peka huruf besar/kecil. Jika kotak centang ini dihapus, pemfilteran klien dilakukan.
4. Pilih Konfirmasi.

Saat mencari berdasarkan tanda, Anda dapat menggunakan nilai berikut:

- (kosong) – Menemukan semua sumber daya dengan kunci tanda yang ditentukan, tetapi tidak boleh ada nilai tanda.
- Semua nilai – Menemukan semua sumber daya dengan kunci tanda yang ditentukan dan nilai tanda apa pun.
- Tidak ditandai – Menemukan semua sumber daya yang tidak memiliki kunci tanda tertentu.
- Nilai yang ditampilkan - Menemukan semua sumber daya dengan kunci tanda tertentu dan nilai tanda tertentu.

Anda dapat menggunakan teknik berikut untuk meningkatkan atau menyempurnakan pencarian:

Pencarian terbalik

Pencarian terbalik memungkinkan Anda mencari sumber daya yang tidak cocok dengan nilai yang ditentukan. Di layar Instans dan AMI, pencarian terbalik dilakukan dengan memilih operator != (Tidak sama dengan) atau !: (Tidak berisi) dan kemudian memilih nilai. Di layar lainnya, pencarian terbalik dilakukan dengan menambahkan prefiks pada kata kunci pencarian dengan karakter tanda seru (!).

> **Note**
>
> Pencarian terbalik didukung dengan pencarian kata kunci dan pencarian atribut hanya pada filter klien. Pencarian ini tidak didukung dengan pencarian atribut pada filter API.

Misalnya, Anda dapat mencari atribut Status instans untuk semua instans guna mengecualikan semua instans yang berada dalam status `terminated`. Untuk melakukannya:

1. Di bidang pencarian pada layar Instans, mulai masukkan `Instance state`. Saat Anda memasukkan karakter, kedua tipe filter muncul untuk Status Instans: Filter API dan Filter Klien.
2. Di bawah Filter klien, pilih Status instans (klien). Pencarian terbalik hanya didukung pada filter klien.

   Daftar operator yang mungkin untuk atribut yang dipilih akan muncul.
3. Pilih != (Tidak sama dengan), lalu pilih diakhiri.

Untuk memfilter instans berdasarkan atribut status instans, Anda juga dapat menggunakan ikon pencarian (



) di kolom Status instans. Ikon pencarian dengan tanda plus ( + ) menampilkan semua instans yang cocok dengan atribut tersebut. Ikon pencarian dengan tanda minus ( - ) mengecualikan semua instans yang cocok dengan atribut tersebut.

Berikut ini contoh lainnya dalam menggunakan pencarian terbalik: Untuk membuat daftar semua instans yang tidak diberikan grup keamanan `launch-wizard-1`, di Filter klien, cari berdasarkan atribut Nama grup keamanan, pilih !=, dan di bilah pencarian, masukkan `launch-wizard-1`.

Pencarian parsial

Dengan pencarian parsial, Anda dapat mencari nilai string parsial. Untuk melakukan pencarian parsial, hanya masukkan sebagian kata kunci yang ingin Anda cari. Pada layar Instans dan AMI, pencarian parsial hanya dapat dilakukan dengan operator: (Berisi). Di layar lainnya, Anda dapat memilih atribut filter klien dan segera memasukkan sebagian kata kunci yang ingin Anda cari saja. Misalnya, pada layar Tipe instans, untuk mencari semua instans `t2.micro`, `t2.small`, dan `t2.medium`, cari berdasarkan atribut Tipe Instans, dan untuk kata kunci, masukkan `t2`.

### Pencarian ekspresi reguler

Untuk menggunakan pencarian ekspresi reguler, Anda harus memilih kotak centang Gunakan pencocokan ekspresi reguler di jendela Preferensi.

Ekspresi reguler berguna saat Anda harus mencocokkan nilai dalam sebuah bidang dengan pola tertentu. Misalnya, untuk mencari nilai yang dimulai dengan `s`, cari `^s`. Untuk mencari nilai yang berakhir dengan `xyz`, cari `xyz$`. Atau, untuk mencari nilai yang dimulai dengan angka yang diikuti oleh satu karakter atau lebih, cari `[0-9]+.*`.

> **Note**
>
> Pencarian ekspresi reguler didukung dengan pencarian kata kunci dan pencarian atribut pada filter klien saja. Pencarian ini tidak didukung dengan pencarian atribut pada filter API.

### Case-sensitive pencarian

Untuk menggunakan pencarian peka huruf besar-kecil, Anda harus memilih kotak centang Gunakan pencocokan peka huruf besar-kecil di jendela P referensi. Preferensi peka huruf besar/kecil hanya berlaku untuk filter klien dan tanda.

> **Note**
>
> Filter API selalu peka huruf besar/kecil.

### Pencarian wildcard

Gunakan wildcard `*` untuk mencocokkan nol atau berbagai karakter. Gunakan wildcard `?` untuk mencocokkan nol atau satu karakter. Misalnya, jika Anda memiliki set data dengan nilai `prod`, `prods`, dan `production`, pencarian `prod*` mencocokkan dengan semua nilai, sedangkan `prod?` hanya mencocokkan `prod` dan `prods`. Untuk menggunakan nilai literal, hindari dengan garis miring terbalik (\). Misalnya, "`prod\*`" akan cocok dengan `prod*`.

> **Note**
>
> Pencarian wildcard didukung dengan pencarian atribut dan tanda pada filter API saja. Pencarian ini tidak didukung dengan pencarian kata kunci, dan dengan pencarian atribut dan tanda pada filter klien.

Pencarian gabungan

Secara umum, banyak filter dengan atribut yang sama secara otomatis digabungkan dengan OR. Misalnya, pencarian `Instance State : Running` dan `Instance State : Stopped` menampilkan semua instans baik yang berjalan ATAU berhenti. Untuk pencarian gabungan dengan AND, cari di berbagai atribut. Misalnya, pencarian `Instance State : Running` dan `Instance Type : c4.large` hanya menampilkan instans dengan tipe `c4.large` DAN yang berada dalam status berjalan.

## Simpan set filter menggunakan konsol

Kum pulan filter yang disimpan adalah grup filter khusus yang dapat Anda buat dan gunakan kembali untuk melihat sumber daya Amazon EC2 secara efisien. Fitur ini membantu merampingkan alur kerja Anda, memungkinkan akses cepat ke tampilan sumber daya tertentu.

Kumpulan filter yang disimpan hanya didukung di konsol Amazon EC2 dan saat ini hanya tersedia untuk halaman Volume.

Fitur utama

- Kustomisasi: Buat set filter yang disesuaikan dengan kebutuhan Anda. Misalnya, Anda dapat membuat set filter untuk hanya menampilkan `gp3` volume yang dibuat setelah tanggal tertentu.
- Filter default: Tetapkan set filter default untuk halaman, dan filter default diterapkan secara otomatis saat Anda menavigasi ke halaman. Jika tidak ada default yang ditetapkan, tidak ada filter yang diterapkan.
- Aplikasi mudah: Pilih set filter yang disimpan untuk menerapkannya secara instan. Amazon EC2 kemudian menampilkan sumber daya yang relevan, dengan filter aktif ditunjukkan oleh token biru.
- Fleksibilitas: Buat, modifikasi, dan hapus set filter sesuai kebutuhan.

### Buat set filter

Untuk membuat set filter baru

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Volume.

> **Note**
>
> Kumpulan filter yang disimpan saat ini hanya tersedia untuk Volume.

3. Di bidang pencarian, pilih filter untuk set filter Anda.
4. Pilih panah di sebelah tombol Hapus filter, dan pilih Simpan set filter baru.
5. Di jendela Simpan set filter, lakukan hal berikut:
   a. Untuk nama set filter, masukkan nama untuk set filter.
   b. (Opsional) Untuk deskripsi set filter, masukkan deskripsi untuk set filter.
   c. (Opsional) Untuk mengatur set filter sebagai filter default, pilih kotak centang Set as default.

   > **Note**
   >
   > Filter default diterapkan secara otomatis setiap kali Anda membuka halaman konsol.

   d. Pilih Simpan.

### Memodifikasi set filter

Untuk memodifikasi kumpulan filter

1. Dari daftar set filter tersimpan, pilih filter yang akan dimodifikasi.
2. Untuk menambahkan filter, di bidang pencarian, pilih filter untuk ditambahkan ke set filter Anda. Untuk menghapus filter dalam set, pilih X pada token filter.
3. Pilih panah di sebelah tombol Hapus filter, dan pilih Ubah set filter.
4. Di jendela Modify filter set, lakukan hal berikut:

a. (Opsional) Untuk mengatur set filter sebagai filter default, pilih kotak centang Set as default.

> **Note**
>
> Filter default diterapkan secara otomatis setiap kali Anda membuka halaman konsol.

b. Pilih Ubah.

Menghapus kumpulan filter

Untuk menghapus kumpulan filter

1. Dari daftar set filter tersimpan, pilih filter yang akan dihapus.
2. Pilih panah di sebelah tombol Hapus filter, dan pilih H apus set filter.
3. Di jendela Hapus set filter, tinjau filter untuk mengonfirmasi bahwa ini adalah filter yang ingin Anda hapus, lalu pilih H apus.

## Daftar dan filter menggunakan baris perintah

Setiap jenis sumber daya memiliki tindakan API yang sesuai yang Anda gunakan untuk mendeskripsikan, mencantumkan, atau mendapatkan sumber daya jenis itu. Daftar sumber daya yang dihasilkan dapat sangat panjang, sehingga lebih cepat dan lebih berguna untuk memfilter hasil guna menyertakan sumber daya yang cocok dengan kriteria tertentu saja.

Pertimbangan pemfilteran

- Anda dapat menentukan hingga 50 filter dan hingga 200 nilai filter total dalam satu permintaan.
- String filter dapat memiliki panjang hingga 255 karakter.
- Anda dapat menggunakan wildcard dengan nilai filter. Tanda bintang (*) cocok dengan nol karakter atau lebih, dan tanda tanya (?) cocok dengan nol atau satu karakter.
- Nilai filter peka huruf besar/kecil.
- Pencarian dapat menyertakan nilai literal dari karakter wildcard; Anda hanya perlu menghindarinya dengan garis miring terbalik sebelum karakter. Misalnya, nilai `\*amazon\?\\` akan mencari string literal `*amazon?\`.

- Anda tidak dapat menentukan nilai filternull. Sebagai gantinya, gunakan pemfilteran sisi klien. Misalnya, perintah berikut menggunakan --query opsi dan mengembalikan ID instance yang diluncurkan tanpa pasangan kunci.

```
aws ec2 describe-instances \
    --query 'Reservations[*].Instances[?!not_null(KeyName)].InstanceId' \
    --output text
```

AWS CLI

Example Contoh: Tentukan satu filter

Anda dapat membuat daftar instans Amazon EC2 menggunakan describe-instances. Tanpa filter, respons akan berisi informasi untuk semua sumber daya Anda. Anda dapat menggunakan opsi berikut untuk menyertakan hanya instance yang sedang berjalan dalam output.

```
--filters Name=instance-state-name,Values=running
```

Untuk mencantumkan hanya ID instance untuk instans yang sedang berjalan, tambahkan `--query` opsi sebagai berikut.

```
--query "Reservations[*].Instances[*].InstanceId"
```

Example Contoh: Tentukan banyak filter atau nilai filter

Jika Anda menentukan beberapa filter atau beberapa nilai filter, sumber daya harus cocok dengan semua filter untuk disertakan dalam output.

Anda dapat menggunakan opsi berikut untuk mencantumkan semua instance yang jenisnya adalah salah satu `m5.large` ataum5d.large.

```
--filters Name=instance-type,Values=m5.large,m5d.large
```

Anda dapat menggunakan opsi berikut untuk mencantumkan semua instance yang dihentikan yang jenisnya adalaht2.micro.

```
--filters Name=instance-state-name,Values=stopped Name=instance-type,Values=t2.micro
```

Example Contoh: Gunakan wildcard dalam nilai filter

Anda dapat menggunakan opsi berikut dengan describe-snapshot untuk mengembalikan hanya snapshot yang deskripsinya adalah “database”.

```
--filters Name=description,Values=database
```

Wildcard * cocok dengan nol karakter atau lebih. Anda dapat menggunakan opsi berikut untuk mengembalikan hanya snapshot yang deskripsinya menyertakan kata database tersebut.

```
--filters Name=description,Values=*database*
```

Wildcard ? cocok dengan 1 karakter saja. Anda dapat menggunakan opsi berikut untuk mengembalikan hanya snapshot yang deskripsinya "" atau database "database" diikuti oleh satu karakter.

```
--filters Name=description,Values=database?
```

Anda dapat menggunakan opsi berikut untuk mengembalikan hanya snapshot yang deskripsinya adalah “database” diikuti hingga empat karakter. Ini tidak termasuk deskripsi dengan “database” diikuti oleh lima karakter atau lebih.

```
--filters Name=description,Values=database????
```

Example Contoh: Filter berdasarkan tanggal

Dengan AWS CLI, Anda dapat menggunakan JMESPath untuk memfilter hasil menggunakan ekspresi. Misalnya, describe-snapshots perintah berikut menampilkan ID semua snapshot yang dibuat oleh yang ditentukan Akun AWS sebelum tanggal yang ditentukan. Jika Anda tidak menentukan pemiliknya, hasilnya akan menyertakan semua snapshot publik.

```
aws ec2 describe-snapshots \
    --filters Name=owner-id,Values=123456789012 \
    --query "Snapshots[?(StartTime<='2024-03-31')].[SnapshotId]" \
    --output text
```

Contoh berikut menampilkan ID semua snapshot yang dibuat dalam rentang tanggal yang ditentukan.

```
aws ec2 describe-snapshots \
    --filters Name=owner-id,Values=123456789012 \
    --query "Snapshots[?(StartTime>='2024-01-01') && (StartTime<='2024-12-31')].
[SnapshotId]" \
    --output text
```

Contoh: Filter berdasarkan tag

Untuk contoh tentang cara memfilter daftar sumber daya menurut tandanya, lihat Filter sumber daya Amazon EC2 berdasarkan tag.

PowerShell

Example Contoh: Tentukan satu filter

Anda dapat membuat daftar instans Amazon EC2 menggunakan Get-EC2Instance. Tanpa filter, respons akan berisi informasi untuk semua sumber daya Anda. Anda dapat menggunakan parameter berikut untuk menyertakan hanya instance yang sedang berjalan dalam output.

```
-Filter @{Name="instance-state-name"; Values="running"}
```

Contoh berikut hanya mencantumkan ID instance untuk instans yang sedang berjalan.

```
(Get-EC2Instance -Filter @{Name="instance-state-name"; Values="stopped"}).Instances
 | Select InstanceId
```

Example Contoh: Tentukan banyak filter atau nilai filter

Jika Anda menentukan banyak filter atau banyak nilai filter, sumber daya harus cocok dengan semua filter yang disertakan dalam hasil.

Anda dapat menggunakan opsi berikut untuk mencantumkan semua instance yang jenisnya adalah salah satu `m5.large` ataum`5d.large`.

```
-Filter @{Name="instance-type"; Values="m5.large", "m5d.large"}
```

Anda dapat menggunakan opsi berikut untuk mencantumkan semua instance yang dihentikan yang jenisnya adalah`t2.micro`.

```
-Filter @{Name="instance-state-name"; Values="stopped"}, @{Name="instance-type";
 Values="t2.micro"}
```

# Melihat sumber daya di seluruh Wilayah menggunakan AWS Tampilan Global

AWS Tampilan Global memungkinkan Anda untuk melihat dan mencari sumber daya di satu Wil AWS ayah, atau di beberapa Wilayah secara bersamaan dalam satu konsol. Lihat informasi yang lebih lengkap di Melihat sumber daya di seluruh Wilayah menggunakan AWS Tampilan Global.

# Melihat sumber daya di seluruh Wilayah menggunakan AWS Tampilan Global

AWS Tampilan Global memungkinkan Anda untuk melihat beberapa sumber daya di satu Wil AWS ayah, atau di beberapa Wilayah dalam satu konsol. AWS Tampilan Global juga menyediakan fungsionalitas pencarian  global yang memungkinkan Anda mencari sumber daya tertentu atau jenis sumber daya tertentu di beberapa Wilayah secara bersamaan.

AWS Tampilan Global tidak mengizinkan Anda memodifikasi sumber daya dengan cara apa pun.

Sumber daya yang didukung

Menggunakan Tampilan AWS Global, Anda dapat melihat ringkasan global dari sumber daya berikut di semua Wilayah yang Anda Akun AWS aktifkan.

- Grup Auto Scaling
- Zona Ketersediaan
- Reservasi Kapasitas dan Blok Kapasitas
- Klaster DB
- Instans DB
- Pengaturan opsi DHCP
- Cluster Layanan Kontainer Elastis Amazon (Amazon ECS)
- Layanan ECS
- Tugas ECS
- Egress-only gerbang internet
- IP Elastis
- Layanan titik akhir
- Instans

- Gateway internet
- Daftar prefiks terkelola
- Gateway NAT
- ACL jaringan
- Antarmuka jaringan
- Outposts
- Tabel rute
- Bucket S3
- Grup keamanan
- Subnet
- Volume
- VPC
- Titik akhir VPC
- Koneksi peering VPC

Izin yang diperlukan

Pengguna harus memiliki izin berikut untuk menggunakan T AWS ampilan Global.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "autoscaling:DescribeAutoScalingGroups",
      "ec2:DescribeRegions",
      "ec2:DescribeCapacityReservations",
      "ec2:DescribeDhcpOptions",
      "ec2:DescribeEgressOnlyInternetGateways",
      "ec2:DescribeAddresses",
      "ec2:DescribeVpcEndpointServices",
      "ec2:DescribeInstances",
      "ec2:DescribeInternetGateways",
```

```
        "ec2:DescribePrefixLists",
        "ec2:DescribeNatGateways",
        "ec2:DescribeNetworkAcls",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeRouteTables",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSubnets",
        "ec2:DescribeVolumes",
        "ec2:DescribeVpcs",
        "ec2:DescribeVpcEndpoints",
        "ec2:DescribeVpcPeeringConnections",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeVpcEndpointServiceConfigurations",
        "ec2:DescribeManagedPrefixLists",
        "ecs:ListClusters",
        "ecs:DescribeServices",
        "ecs:ListTasks",
        "outposts:ListOutposts",
        "rds:DescribeDBInstances",
        "rds:DescribeDBClusters",
        "s3:ListAllMyBuckets"
    ],
    "Resource": "*"
    }]
}
```

Untuk menggunakan AWS Tampilan Global

Masuk ke konsol AWS Tampilan Global.

> **Important**
>
> Anda tidak dapat menggunakan jendela pribadi di Firefox untuk mengakses T AWS ampilan Global.

Konsol tersebut terdiri dari hal-hal berikut:

- Penjelajah wilayah - Halaman ini mencakup bagian-bagian berikut:
  - Ringkasan — Menyediakan ikhtisar tingkat tinggi sumber daya Anda di semua Wilayah.

Perluas Tampilkan semua ringkasan sumber daya menunjukkan jumlah Wilayah Akun AWS yang Anda aktifkan. Bidang yang tersisa menunjukkan jumlah sumber daya yang saat ini Anda miliki di Wilayah tersebut. Pilih salah satu tautan untuk melihat sumber daya tipe tersebut di semua Wilayah. Misalnya, jika tautan di bawah label Instans 29 di 10 Wilayah, hal ini menunjukkan bahwa saat ini Anda memiliki 29 instans di 10 Wilayah. Pilih tautan untuk melihat daftar 29 instans.

- Penjelaj ah wilayah — Mencantumkan semua Wilayah AWS (termasuk yang akun Anda tidak diaktifkan) dan memberikan total untuk setiap jenis sumber daya untuk setiap Wilayah.

  Pilih nama Wilayah untuk melihat semua sumber daya dari semua tipe untuk Wilayah tertentu. Misalnya, pilih Afrika (Cape Town) af-south-1 untuk melihat semua VPC, subnet, instans, grup keamanan, volume, dan grup Auto Scaling di Wilayah tersebut. Atau, pilih Wilayah dan pilih Lihat sumber daya untuk Wilayah yang dipilih.

  Pilih nilai untuk tipe sumber daya tertentu di Wilayah tertentu untuk hanya melihat sumber daya dari tipe tersebut di Wilayah tersebut. Misalnya, pilih nilai untuk Instans Afrika (Cape Town) af-south-1 untuk hanya melihat instans di Wilayah tersebut.

- Pencarian global — Halaman ini memungkinkan Anda untuk mencari sumber daya tertentu atau jenis sumber daya tertentu di satu Wilayah atau di beberapa Wilayah. Tab tersebut juga memungkinkan Anda melihat detail sumber daya tertentu.

  Untuk mencari sumber daya, masukkan kriteria pencarian di bidang sebelum grid. Anda dapat mencari berdasarkan Wilayah, berdasarkan tipe sumber daya, dan berdasarkan tanda yang ditetapkan ke sumber daya.

  Untuk melihat detail sumber daya tertentu, pilih sumber daya tersebut di grid. Anda juga dapat memilih ID sumber daya dari sebuah sumber daya untuk membukanya di konsol masing-masing. Misalnya, pilih ID instans untuk membuka instans di konsol Amazon EC2, atau pilih ID subnet untuk membuka subnet di konsol Amazon VPC.

- Wilayah dan Zona — Halaman ini memungkinkan Anda untuk melihat dan mengelola semua Wilayah yang tersedia, Zona Ketersediaan, Zona Lokal, dan Zona Panjang Gelombang. Anda dapat beralih antara tampilan Daftar dan tampilan Peta menggunakan sakelar Tampilan daftar / tampilan peta. Preferensi tampilan Anda tetap ada di seluruh sesi.
  - Tampilan peta - Menampilkan semua Wilayah dan Zona pada peta dunia interaktif, diwarnai berdasarkan status. Untuk melihat detail lokasi, arahkan ke lokasi atau pilih. Pilih lokasi untuk mengambil tindakan — Anda dapat mengaktifkan Wilayah, mengaktifkan Zona, atau membuka

halaman detailnya. Anda juga dapat memfilter peta untuk hanya menampilkan status tertentu. Peta dan daftar detail tetap disinkronkan — memilih lokasi di satu menyorotnya di yang lain.

- Tampilan daftar - Menampilkan Wilayah dan Zona dalam tabel. Dari tab Wilayah, Anda dapat melihat semua Wil AWS ayah. Kol om Status menunjukkan Wilayah yang diaktifkan untuk Anda Akun AWS.

Dari halaman ini, Anda dapat memilih Wilayah untuk:

- Lihat detail Wilayah seperti kode Wilayah, Geografi, dan jumlah setiap jenis Zona.

  Anda juga dapat melihat daftar Zona Ketersediaan, Zona Lokal, dan Zona Panjang Gelombang dan daftar sumber daya Anda di Wilayah.
- Aktifkan atau menonaktifkan Wilayah.

Dari setiap tab zona, Anda dapat melihat daftar jenis zona tersebut. Dari tab Zona Lokal, Anda dapat memilih untuk masuk ke Zona Lokal.

> **Tip**
>
> Jika Anda hanya menggunakan Wilayah atau jenis sumber daya tertentu, Anda dapat menyesuaikan T AWS ampilan Global untuk menampilkan hanya Wilayah dan jenis sumber daya tersebut. Untuk menyesuaikan Wilayah dan jenis sumber daya yang ditampilkan, di panel navigasi, pilih Peng aturan, lalu pada tab Sumber Daya dan Wil ayah, pilih Wilayah dan jenis sumber daya yang tidak ingin ditampilkan di T AWS ampilan Global.

# Tandai sumber daya Amazon EC2 Anda

Untuk membantu mengelola instans, citra, dan sumber daya Amazon EC2 lainnya, Anda dapat menetapkan metadata Anda sendiri ke setiap sumber daya dalam bentuk tanda. Tag memungkinkan Anda untuk mengkategorikan AWS sumber daya Anda dengan cara yang berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Hal ini berguna ketika Anda memiliki banyak sumber daya dengan jenis yang sama—Anda dapat dengan cepat mengidentifikasi sumber daya tertentu berdasarkan tag yang telah Anda tetapkan. Topik ini menjelaskan tanda dan menunjukkan cara membuatnya.

> **Warning**
>
> Kunci tanda dan nilainya akan ditampilkan oleh berbagai panggilan API. Menolak akses ke `DescribeTags` tidak secara otomatis menolak akses ke tanda yang ditampilkan oleh API lain. Sebagai praktik terbaik, sebaiknya Anda tidak menyertakan data sensitif ke dalam tanda.

Daftar Isi

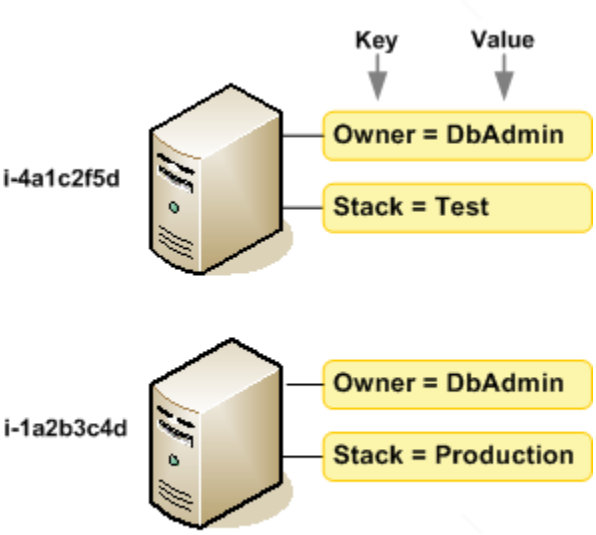

- Dasar-dasar tag
- Tandai sumber daya Anda
- Pembatasan tanda
- Manajemen tanda dan akses
- Menandai sumber daya Anda untuk penagihan
- Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan
- Menambahkan dan menghapus tag untuk sumber daya Amazon EC2
- Filter sumber daya Amazon EC2 berdasarkan tag
- Melihat tag untuk instans EC2 menggunakan metadata instans


## Dasar-dasar tag

Tag adalah label yang Anda tetapkan ke AWS sumber daya. Setiap tag terdiri atas sebuah kunci dan sebuah nilai opsional, dan Anda menentukan sendiri keduanya.

Tag memungkinkan Anda untuk mengkategorikan AWS sumber daya Anda dengan cara yang berbeda, misalnya, berdasarkan tujuan, pemilik, atau lingkungan. Misalnya, Anda dapat menentukan satu set tanda untuk instans Amazon EC2 dari akun Anda yang dapat membantu melacak setiap pemilik dan tingkat tumpukan instans.

Diagram berikut menggambarkan cara kerja penandaan. Dalam contoh ini, Anda telah menetapkan dua tanda pada setiap instans—satu tanda dengan kunci `Owner` dan tanda lain dengan kunci `Stack`. Setiap tanda juga memiliki nilai yang terkait.




Sebaiknya rancang serangkaian kunci tag yang memenuhi kebutuhan setiap jenis sumber daya. Penggunaan set kunci tag yang konsisten akan memudahkan manajemen sumber daya Anda. Anda dapat mencari dan memfilter sumber daya berdasarkan tag yang Anda tambahkan. Untuk informasi selengkapnya tentang cara menerapkan strategi penandaan sumber daya yang efektif, lihat Whitepaper Prakti k Terbaik Menandai. AWS

Tanda tidak memiliki makna semantik untuk Amazon EC2 dan diinterpretasikan sebagai string karakter. Selain itu, tanda tidak secara otomatis ditetapkan ke sumber daya Anda. Anda dapat mengedit kunci dan nilai tanda, dan dapat menghapus tanda dari sumber daya kapan saja. Anda dapat mengatur nilai tanda ke string kosong, tetapi tidak dapat mengatur nilai tanda ke null. Jika Anda menambahkan tanda yang memiliki kunci yang sama dengan tanda yang telah ada pada sumber daya tersebut, nilai yang baru akan menimpa nilai yang lama. Jika Anda menghapus sumber daya, tanda apa pun untuk sumber daya tersebut juga dihapus.

> **Note**
>
> Setelah Anda menghapus sumber daya, tanda sumber daya tersebut mungkin tetap terlihat di konsol, API, dan output CLI untuk waktu yang singkat. Tanda ini akan secara bertahap dipisahkan dari sumber daya dan dihapus secara permanen.

# Tandai sumber daya Anda

Saat menggunakan konsol Amazon EC2, Anda dapat menerapkan tag ke sumber daya dengan menggunakan tab Tag pada layar sumber daya yang relevan, atau Anda dapat menggunakan Editor Tag di AWS Resource Groups konsol. Beberapa layar sumber daya memungkinkan Anda menentukan tanda untuk sebuah sumber daya saat sumber daya tersebut dibuat; misalnya, tanda dengan kunci `Name` dan nilai yang Anda tentukan. Dalam kebanyakan kasus, konsol menerapkan tanda segera setelah sumber daya dibuat (alih-alih selama pembuatan sumber daya). Konsol dapat mengatur sumber daya sesuai dengan tanda `Name`, tetapi tanda ini tidak memiliki makna semantik untuk layanan Amazon EC2.

Jika Anda menggunakan Amazon EC2 API, atau AWS SDK AWS CLI, Anda dapat menggunakan tindakan API `CreateTags` EC2 untuk menerapkan tag ke sumber daya yang ada. Selain itu, beberapa tindakan pembuatan sumber daya memungkinkan Anda menentukan tag untuk sumber daya saat sumber daya tersebut dibuat. Jika tag tidak dapat diterapkan selama pembuatan sumber daya, kami akan mengembalikan proses pembuatan sumber daya. Hal ini untuk memastikan bahwa sumber daya dibuat dengan tag atau tidak akan dibuat sama sekali, dan tidak akan ada sumber daya yang dibiarkan tidak bertanda. Dengan menandai sumber daya saat pembuatan, Anda dapat menghilangkan kebutuhan untuk menjalankan skrip penandaan kustom setelah pembuatan sumber daya. Untuk informasi selengkapnya tentang memungkinkan pengguna menandai sumber daya saat pembuatan, lihat Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan.

Anda dapat menerapkan izin tingkat sumber daya berbasis tanda dalam kebijakan IAM untuk tindakan API Amazon EC2 yang mendukung penandaan saat pembuatan guna mengimplementasikan kontrol terperinci atas pengguna dan grup yang dapat menandai sumber daya saat pembuatan. Sumber daya Anda diamankan secara tepat sejak pembuatan—tanda segera diterapkan pada sumber daya Anda, oleh karena itu, izin tingkat sumber daya berbasis tanda yang mengontrol penggunaan sumber daya langsung berlaku. Sumber daya Anda dapat dilacak dan dilaporkan dengan lebih akurat. Anda dapat menerapkan penggunaan penandaan pada sumber daya baru serta mengontrol kunci dan nilai tanda mana yang ditetapkan pada sumber daya Anda.

Anda juga dapat menerapkan izin tingkat sumber daya untuk tindakan API Amazon EC2 `CreateTags` dan `DeleteTags` dalam kebijakan IAM Anda untuk mengontrol kunci dan nilai tanda mana yang ditetapkan pada sumber daya yang ada. Untuk informasi selengkapnya, lihat Contoh: Memberi tanda pada sumber daya.

Untuk informasi selengkapnya tentang penandaan sumber daya untuk penagihan, lihat Menggunakan tanda alokasi biaya dalam Buku Panduan AWS Billing .

# Pembatasan tanda

Batasan dasar berikut berlaku untuk tag:

- Jumlah maksimum tag per sumber daya – 50
- Untuk setiap sumber daya, setiap kunci tag harus unik, dan setiap kunci tag hanya dapat memuat satu nilai.
- Panjang kunci maksimum - 128 karakter Unicode di UTF-8
- Panjang nilai maksimum - 256 karakter Unicode di UTF-8
- Karakter yang diizinkan
  - Meskipun EC2 memungkinkan karakter apa pun dalam tagnya, AWS layanan lain lebih membatasi. Karakter yang diizinkan di semua AWS layanan adalah: huruf (`a-z`,`A-Z`), angka (`0-9`), dan spasi yang dapat direpresentasikan dalam UTF-8, dan karakter berikut:`+ - = . _ : / @`.
  - Jika Anda mengaktifkan tanda instans dalam metadata instans, kunci tanda instans hanya dapat menggunakan huruf (`a-z`, `A-Z`), angka (`0-9`), dan karakter berikut: `+ - = . , _ : @`. Kunci tanda instans tidak dapat memuat spasi atau `/`, dan tidak dapat terdiri dari `.` (satu titik), `..` (dua titik), atau `_index` saja. Untuk informasi selengkapnya, lihat Melihat tag untuk instans EC2 menggunakan metadata instans.
- Kunci dan nilai tanda peka huruf besar/kecil.
- A `aws:` walan dicadangkan untuk AWS digunakan. Jika tag memiliki kunci tag dengan awalan ini, Anda tidak dapat mengedit atau menghapus kunci atau nilai tag tersebut. Tag dengan awalan `aws:` tidak dihitung terhadap tag per batas sumber daya.

Anda tidak dapat mengakhiri, menghentikan, atau menghapus sumber daya berdasarkan tandanya saja; Anda harus menentukan pengidentifikasi sumber daya tersebut. Misalnya, untuk menghapus snapshot yang Anda beri tag dengan tag kunci yang disebut `DeleteMe`, Anda harus menggunakan tindakan `DeleteSnapshots` dengan pengidentifikasi sumber daya snapshot tersebut, seperti `snap-1234567890abcdef0`.

Saat Anda menandai sumber daya publik atau bersama, tag yang Anda tetapkan hanya tersedia untuk AWS akun Anda; tidak ada AWS akun lain yang memiliki akses ke tag tersebut. Untuk kontrol akses berbasis tag ke sumber daya bersama, setiap AWS akun harus menetapkan set tag sendiri untuk mengontrol akses ke sumber daya.

## Manajemen tanda dan akses

Jika Anda menggunakan AWS Identity and Access Management (IAM), Anda dapat mengontrol pengguna mana di AWS akun Anda yang memiliki izin untuk membuat, mengedit, atau menghapus tag. Untuk informasi selengkapnya, lihat Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan.

Anda dapat menggunakan tanda sumber daya untuk mengimplementasikan kontrol berbasis atribut (ABAC). Anda dapat membuat kebijakan IAM yang memungkinkan operasi berdasarkan tanda untuk sumber daya. Untuk informasi selengkapnya, lihat Kontrol akses menggunakan akses berbasis atribut.

## Menandai sumber daya Anda untuk penagihan

Anda dapat menggunakan tag untuk mengatur AWS tagihan agar mencerminkan struktur biaya Anda sendiri. Untuk melakukan ini, daftar untuk mendapatkan tagihan AWS akun Anda dengan nilai kunci tag disertakan. Untuk informasi selengkapnya tentang pengaturan laporan alokasi biaya dengan tanda, lihat Laporan alokasi biaya bulanan di Panduan Pengguna AWS Billing . Untuk melihat biaya sumber daya gabungan, Anda dapat mengatur informasi penagihan berdasarkan sumber daya yang memiliki nilai kunci tanda yang sama. Misalnya, Anda dapat menandai beberapa sumber daya dengan nama aplikasi tertentu, kemudian mengatur informasi penagihan untuk melihat biaya total aplikasi tersebut pada beberapa layanan. Untuk informasi selengkapnya, lihat Menggunakan tanda alokasi biaya dalam Panduan Pengguna AWS Billing .

> **Note**
>
> Jika Anda baru saja mengaktifkan pelaporan, data untuk bulan yang berjalan dapat dilihat setelah 24 jam.

Tanda alokasi biaya dapat mengindikasikan sumber daya mana yang memengaruhi biaya, tetapi penghapusan atau penonaktifkan sumber daya tidak selalu mengurangi biaya. Misalnya, data snapshot yang direferensikan oleh snapshot lain disimpan, bahkan jika snapshot yang berisi data asli dihapus. Untuk informasi selengkapnya, lihat Snapshot dan volume Amazon Elastic Block Store di Panduan Pengguna AWS Billing .

> **Note**
>
> Alamat IP Elastis yang diberikan tanda tidak akan muncul pada laporan alokasi biaya Anda.

## Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan

Beberapa tindakan API Amazon EC2 yang digunakan untuk membuat sumber daya memungkinkan Anda menentukan tanda saat Anda membuat sumber daya. Anda dapat menggunakan tanda sumber daya untuk menerapkan pengendalian berbasis atribut (ABAC). Untuk informasi selengkapnya, lihat Tandai sumber daya Anda dan Kontrol akses menggunakan akses berbasis atribut.

Untuk memungkinkan para pengguna memberikan tanda pada sumber daya pada saat pembuatan, para pengguna tersebut harus memiliki izin untuk menggunakan tindakan-tindakan yang membuat sumber daya, seperti `ec2:RunInstances` atau `ec2:CreateVolume`. Jika tanda-tanda ditentukan dalam tindakan yang digunakan untuk membuat sumber daya, maka Amazon akan melakukan otorisasi tambahan pada tindakan `ec2:CreateTags` untuk melakukan verifikasi apakah pengguna memiliki izin untuk membuat tanda. Oleh karena itu, para pengguna juga harus memiliki izin eksplisit untuk menggunakan tindakan `ec2:CreateTags`.

Di dalam definisi kebijakan IAM untuk tindakan `ec2:CreateTags`, gunakan elemen `Condition` dengan kunci syarat `ec2:CreateAction` untuk memberikan izin pemberian tanda pada tindakan yang membuat sumber daya.

Contoh berikut ini mendemonstrasikan kebijakan yang memungkinkan para pengguna untuk meluncurkan instans dan menerapkan tanda apa pun pada instans dan volume saat dilakukan peluncuran. Pengguna tidak diizinkan untuk menandai sumber daya yang sudah ada (mereka tidak dapat memanggil tindakan `ec2:CreateTags` secara langsung).

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:RunInstances"
      ],
      "Resource": "*"
    },
    {
```

```
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:*/*",
      "Condition": {
         "StringEquals": {
             "ec2:CreateAction" : "RunInstances"
          }
       }
    }
  ]
}
```

Demikian pula, kebijakan berikut memungkinkan para pengguna untuk membuat volume dan menerapkan tanda apa pun pada volume saat volume dibuat. Para pengguna tidak diizinkan untuk memberi tanda pada sumber daya yang sudah ada (mereka tidak dapat memerintahkan tindakan `ec2:CreateTags` secara langsung).

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateVolume"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "ec2:CreateTags"
      ],
      "Resource": "arn:aws:ec2:us-east-1:111122223333:*/*",
      "Condition": {
         "StringEquals": {
             "ec2:CreateAction" : "CreateVolume"
          }
       }
    }
  ]
}
```

Tindakan `ec2:CreateTags` akan dievaluasi hanya jika tanda diterapkan selama tindakan pembuatan sumber daya. Oleh karena itu, seorang pengguna yang memiliki izin untuk membuat sumber daya (dengan asumsi tidak ada syarat untuk pemberian tanda) tidak memerlukan izin untuk menggunakan tindakan `ec2:CreateTags` jika tidak ada tanda yang ditentukan dalam permintaan. Akan tetapi, jika pengguna tersebut mencoba untuk membuat sumber daya dengan tanda, maka permintaan akan gagal jika pengguna tidak memiliki izin untuk menggunakan tindakan `ec2:CreateTags`.

Tindakan `ec2:CreateTags` juga akan dievaluasi jika tanda disediakan dalam templat peluncuran. Untuk melihat contoh kebijakan IAM, lihat Tanda di templat peluncuran.

## Mengendalikan akses ke tanda-tanda tertentu

Anda dapat menggunakan syarat tambahan dalam elemen `Condition` dari kebijakan IAM Anda untuk mengontrol kunci tanda dan nilai tanda yang dapat diterapkan ke sumber daya.

Kunci syarat berikut dapat digunakan dengan contoh-contoh pada bagian sebelumnya:

- `aws:RequestTag`: Untuk mengindikasikan bahwa kunci tanda tertentu atau kunci dan nilai tanda tertentu harus ada di permintaan. Tanda-tanda yang lain juga dapat ditentukan dalam permintaan.
  - Gunakan bersama dengan operator syarat `StringEquals` untuk memberlakukan kombinasi kunci dan nilai tanda tertentu, misalnya, untuk memberlakukan tanda `cost-center=cc123`:

    ```
    "StringEquals": { "aws:RequestTag/cost-center": "cc123" }
    ```

  - Gunakan bersama dengan operator syarat `StringLike` untuk memberlakukan kunci tanda tertentu dalam permintaan, misalnya, untuk memberlakukan kunci tanda `purpose`:

    ```
    "StringLike": { "aws:RequestTag/purpose": "*" }
    ```

- `aws:TagKeys`: Untuk memberlakukan kunci tanda yang digunakan dalam permintaan.
  - Gunakan bersama dengan pemodifikasi `ForAllValues` untuk menerapkan kunci tanda tertentu jika disediakan dalam permintaan (jika tanda ditentukan dalam permintaan, hanya kunci tanda tertentu saja yang diperbolehkan; tidak ada tanda lain yang diperbolehkan). Sebagai contoh, kunci tanda `environment` atau `cost-center` diperbolehkan:

    ```
    "ForAllValues:StringEquals": { "aws:TagKeys": ["environment","cost-center"] }
    ```

- Gunakan pemodifikasi `ForAnyValue` untuk memaksakan keberadaan setidaknya salah satu kunci tanda tertentu dalam permintaan. Sebagai contoh, setidaknya salah satu kunci tanda `environment` atau `webserver` harus ada dalam permintaan:

```
"ForAnyValue:StringEquals": { "aws:TagKeys": ["environment","webserver"] }
```

Kunci syarat ini dapat diterapkan untuk tindakan-tindakan yang digunakan untuk membuat sumber daya yang mendukung pemberian tanda, serta tindakan `ec2:CreateTags` dan `ec2:DeleteTags`. Untuk mempelajari apakah tindakan API Amazon EC2 mendukung pemberian tanda, lihat Tindakan, sumber daya, dan kunci syarat untuk Amazon EC2.

Untuk memaksa para pengguna menentukan tanda pada saat mereka membuat sumber daya, Anda harus menggunakan kunci syarat `aws:RequestTag` atau kunci syarat `aws:TagKeys` dengan pemodifikasi `ForAnyValue` pada tindakan yang digunakan untuk membuat sumber daya. Tindakan `ec2:CreateTags` tidak akan dievaluasi jika pengguna tidak menentukan tanda untuk tindakan yang digunakan untuk pembuatan sumber daya.

Untuk syarat, kunci syarat tidak bersifat peka terhadap huruf besar dan kecil dan nilai syarat bersifat peka huruf besar dan kecil. Oleh karena itu, untuk memaksakan sifat peka terhadap huruf besar atau kecil dari kunci tanda, gunakan kunci syarat `aws:TagKeys`, di mana kunci tanda ditetapkan sebagai nilai dalam syarat tersebut.

Untuk contoh kebijakan IAM, lihat Contoh kebijakan untuk mengontrol akses Amazon EC2 API. Untuk informasi selengkapnya, lihat Kondisi dengan beberapa kunci konteks atau nilai di Panduan Pengguna IAM.

## Menambahkan dan menghapus tag untuk sumber daya Amazon EC2

Saat membuat sumber daya Amazon EC2, seperti instans Amazon EC2, Anda dapat menentukan tag yang akan ditambahkan ke sumber daya. Anda juga dapat menggunakan konsol Amazon EC2 untuk menampilkan tag untuk sumber daya Amazon EC2 tertentu. Anda juga dapat menambahkan atau menghapus tag dari sumber daya Amazon EC2 yang ada.

Anda dapat menggunakan Editor Tag di AWS Resource Groups konsol untuk melihat, menambah, atau menghapus tag di semua sumber AWS daya Anda di semua Wilayah. Anda dapat menerapkan atau menghapus tag dari beberapa jenis sumber daya secara bersamaan. Untuk informasi selengkapnya, lihat Panduan Pengguna Sumber AWS Daya Tagging.

Tugas

- Tambahkan tag menggunakan konsol
- Tambahkan tag menggunakan AWS CLI
- Tambahkan tag menggunakan PowerShell
- Tambahkan tag menggunakan CloudFormation

## Tambahkan tag menggunakan konsol

Anda dapat menambahkan tag ke sumber daya yang ada langsung dari halaman untuk sumber daya.

Untuk menambahkan tag ke sumber daya yang ada

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dari bilah navigasi, pilih Wilayah tempat sumber daya berada.
3. Di panel navigasi, pilih tipe sumber daya (misalnya, Instans).
4. Pilih sumber daya dari daftar.
5. Dari tab Tag, pilih Kel ola tag.
6. Pilih Tambahkan tag baru dan masukkan kunci dan nilai untuk tag.
7. Pilih Simpan.

## Tambahkan tag menggunakan AWS CLI

Anda dapat menambahkan tag saat membuat sumber daya atau ke sumber daya yang ada.

Untuk menambahkan tag pada pembuatan sumber daya

Gunakan `-tag-specifications` opsi untuk menandai sumber daya pada pembuatan. Spesifikasi tag memerlukan jenis sumber daya yang akan diberi tag, kunci tag, dan nilai tag. Contoh berikut membuat tag dan menambahkannya ke spesifikasi tag.

```
--tag-specifications 'ResourceType=instance,Tags=[{Key=stack,Value=production}]'
```

Untuk menambahkan tag ke sumber daya yang ada

Contoh berikut menunjukkan cara menambahkan tag ke sumber daya yang ada menggunakan perintah create-tags.

Example Contoh: Menambahkan tanda ke sumber daya

Perintah berikut menambahkan tanda **`Stack=production`** ke citra tertentu, atau menimpa tanda yang sudah ada untuk AMI di mana kunci tandanya adalah stack. Jika perintah berhasil, tidak ada output yang akan ditampilkan.

```
aws ec2 create-tags \
    --resources ami-0abcdef1234567890 \
    --tags Key=stack,Value=production
```

Example Contoh: Menambahkan tanda ke banyak sumber daya

Contoh ini menambahkan (atau menimpa) dua tanda untuk AMI dan instans. Salah satu tanda hanya berisi kunci (webserver), dan tanpa nilai (kami mengatur nilai ke string kosong). Tanda lainnya terdiri dari kunci (stack) dan nilai (**`Production`**). Jika perintah berhasil, tidak ada output yang akan ditampilkan.

```
aws ec2 create-tags \
    --resources ami-0abcdef1234567890 i-1234567890abcdef0 \
    --tags Key=webserver,Value=  Key=stack,Value=Production
```

Example Contoh: Menambahkan tanda dengan karakter khusus

Contoh ini menambahkan tanda [Group]=test ke instans. Tanda kurung siku ([ dan ]) adalah karakter khusus, yang harus dihindari.

Jika Anda menggunakan Linux atau OS X, untuk mengecualikan karakter khusus, sertakan elemen dengan karakter khusus dengan petik ganda (''), lalu sertakan seluruh kunci dan struktur nilai dengan tanda petik tunggal (').

```
aws ec2 create-tags \
    --resources i-1234567890abcdef0 \
    --tags 'Key="[Group]",Value=test'
```

Jika Anda menggunakan Windows, untuk mengecualikan karakter khusus, sertakan elemen yang memiliki karakter khusus dengan petik ganda (“), lalu di depan setiap karakter bertanda petik ganda, tambahkan garis miring terbalik (****) sebagai berikut:

```
aws ec2 create-tags ^
```

```
    --resources i-1234567890abcdef0 ^
    --tags Key=\"[Group]\",Value=test
```

Jika Anda menggunakan Windows PowerShell, untuk menghindari karakter khusus, lampirkan nilai yang memiliki karakter khusus dengan tanda kutip ganda (**"**), mendahului setiap karakter kutipan ganda dengan garis miring terbalik (****), lalu lampirkan seluruh struktur kunci dan nilai dengan tanda kutip tunggal () sebagai berikut: **'**

```
aws ec2 create-tags `
    --resources i-1234567890abcdef0 `
    --tags 'Key=\"[Group]\",Value=test'
```

## Tambahkan tag menggunakan PowerShell

Anda dapat menambahkan tag saat membuat sumber daya atau ke sumber daya yang ada.

Untuk menambahkan tag pada pembuatan sumber daya

Gunakan `-TagSpecification` parameter untuk menandai sumber daya pada pembuatan. Spesifikasi tag memerlukan jenis sumber daya yang akan diberi tag, kunci tag, dan nilai tag. Contoh berikut membuat tag dan menambahkannya ke spesifikasi tag.

```
$tag = @{Key="stack"; Value="production"}
$tagspec = new-object Amazon.EC2.Model.TagSpecification
$tagspec.ResourceType = "instance"
$tagspec.Tags.Add($tag)
```

Contoh berikut menentukan tag ini dalam `-TagSpecification` parameter.

```
-TagSpecification $tagspec
```

Untuk menambahkan tag ke sumber daya yang ada

Gunakan New-EC2Tag cmdlet. Anda harus menentukan sumber daya, kunci tag, dan nilai tag.

```
New-EC2Tag `
    -Resource i-1234567890abcdef0 `
    -Tag @{Key="purpose"; Value="production"}
```

## Tambahkan tag menggunakan CloudFormation

Dengan tipe sumber daya Amazon EC2, Anda menentukan tanda menggunakan `Tags` atau properti `TagSpecifications`.

Contoh berikut menambahkan tanda **`Stack=Production`** pada AWS::EC2::Instance menggunakan properti `Tags`.

Example Contoh: Tanda dalam YAML

```
Tags:
  - Key: "Stack"
    Value: "Production"
```

Example Contoh: Tanda dalam JSON

```
"Tags": [
    {
        "Key": "Stack",
        "Value": "Production"
    }
]
```

Contoh berikut menambahkan tag **`Stack=Production`** ke AWS: :EC2:: LaunchTemplate LaunchTemplateData menggunakan proper `TagSpecifications` tinya.

Example Contoh: TagSpecifications di YAML

```
TagSpecifications:
  - ResourceType: "instance"
    Tags:
    - Key: "Stack"
      Value: "Production"
```

Example Contoh: TagSpecifications dalam JSON

```
"TagSpecifications": [
    {
        "ResourceType": "instance",
        "Tags": [
            {
```

```
                "Key": "Stack",
                "Value": "Production"
            }
        ]
    }
]
```

## Filter sumber daya Amazon EC2 berdasarkan tag

Setelah menambahkan tag, Anda dapat memfilter kunci tag dan nilai tag berdasarkan sumber daya Amazon EC2.

Console

Untuk memfilter sumber daya berdasarkan tag

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih tipe sumber daya (misalnya, Instans).
3. Pilih bidang pencarian.
4. Dalam daftar, di bawah Tanda, pilih kunci tanda.
5. Pilih nilai tanda yang sesuai dari daftar.
6. Setelah selesai, hapus filter.

Untuk informasi selengkapnya tentang penggunaan filter di konsol Amazon EC2, lihat Temukan sumber daya Amazon EC2 Anda.

AWS CLI

Untuk menjelaskan sumber daya dari satu jenis dengan kunci tag yang ditentukan

Tambahkan filter berikut ke `describe` perintah untuk menggambarkan sumber daya jenis itu dengan Stack tag, terlepas dari nilai tag.

```
--filters Name=tag-key,Values=Stack
```

Untuk mendeskripsikan sumber daya dari satu jenis dengan tag yang ditentukan

Tambahkan filter berikut ke `describe` perintah untuk menggambarkan sumber daya jenis itu dengan tagStack=production.

```
--filters Name=tag:Stack,Values=production
```

Untuk mendeskripsikan sumber daya dari satu jenis dengan nilai tag yang ditentukan

Tambahkan filter berikut ke `describe` perintah untuk menggambarkan sumber daya jenis itu dengan tag dengan nilaiproduction, terlepas dari kunci tag.

```
--filters Name=tag-value,Values=production
```

Untuk menjelaskan semua sumber daya EC2 dengan tag yang ditentukan

Tambahkan filter berikut ke perintah describe-tags untuk menjelaskan semua sumber daya EC2 dengan tag. Stack=test

```
--filters Name=key,Values=Stack Name=value,Values=test
```

PowerShell

Untuk memfilter sumber daya dari satu jenis dengan kunci tag

Tambahkan filter berikut ke `Get` cmdlet untuk menggambarkan sumber daya jenis itu dengan Stack tag, terlepas dari nilai tag.

```
-Filter @{Name="tag-key"; Values="Stack"}
```

Untuk memfilter sumber daya dari satu jenis berdasarkan tag

Tambahkan filter berikut ke `Get` cmdlet untuk menggambarkan sumber daya jenis itu dengan tag. Stack=production

```
-Filter @{Name="tag:Stack"; Values="production"}
```

Untuk memfilter sumber daya dari satu jenis berdasarkan nilai tag

Tambahkan filter berikut ke `Get` cmdlet untuk menggambarkan sumber daya jenis itu dengan tag dengan nilaiproduction, terlepas dari nilai kunci tag.

```
-Filter @{Name="tag-value"; Values="production"}
```

Untuk memfilter semua sumber daya EC2 berdasarkan tag

Tambahkan filter berikut ke Get-EC2Tag cmdlet untuk menjelaskan semua sumber daya EC2 dengan tag. Stack=test

```
-Filter @{Name="tag:Stack"; Values="test"}
```

## Melihat tag untuk instans EC2 menggunakan metadata instans

Anda dapat mengakses tanda instans dari metadata instans. Dengan mengakses tanda dari metadata instans, Anda tidak perlu lagi menggunakan panggilan API `DescribeInstances` atau `DescribeTags` untuk mengambil informasi tanda, yang mengurangi transaksi API per detik, dan memungkinkan pengambilan tanda diskalakan dengan jumlah instans yang Anda kontrol. Selain itu, proses lokal yang berjalan pada sebuah instans dapat melihat informasi tanda instans secara langsung dari metadata instans.

Secara default, tanda tidak tersedia dari metadata instans; Anda harus secara eksplisit mengizinkan akses. Anda dapat mengizinkan akses saat peluncuran instans, atau setelah peluncuran pada instans yang sedang berjalan atau dihentikan. Anda juga dapat mengizinkan akses ke tanda dengan menentukannya dalam templat peluncuran. Instans yang diluncurkan menggunakan templat mengizinkan akses ke tanda dalam metadata instans.

Jika Anda menambahkan atau menghapus tanda instans, metadata instans diperbarui saat instans sedang berjalan, tanpa perlu berhenti dan kemudian memulai instans.

Tugas

- Aktifkan akses ke tag dalam metadata instance
- Mengambil tanda dari metadata instans
- Nonaktifkan akses ke tag dalam metadata instance

### Aktifkan akses ke tag dalam metadata instance

Secara default, tidak ada akses ke tanda instans dalam metadata instans. Untuk setiap contoh, Anda harus secara eksplisit mengaktifkan akses.

> **Note**
>
> Jika Anda mengizinkan akses ke tag dalam metadata instance, kunci tag instance tunduk pada batasan tertentu. Non-compliance akan mengakibatkan peluncuran gagal untuk instance baru atau kesalahan untuk instance yang ada. Batasannya adalah:
>
> - Hanya dapat menyertakan huruf (`a-z`,`A-Z`), angka (`0-9`), dan karakter berikut:`+ - = . , _ : @`.
> - Tidak dapat berisi spasi atau`/`.
> - Tidak bisa hanya terdiri dari `.` (satu periode), `..` (dua periode), atau`_index`.
>
> Untuk informasi selengkapnya, lihat Pembatasan tanda.

Console

Untuk mengaktifkan akses ke tag dalam metadata instance selama peluncuran instance

1. Ikuti prosedur untuk meluncurkan instans.
2. Perluas Detail lanj utan, dan untuk Izinkan tag dalam metadata, pilih Aktifkan.
3. Di panel Summary, tinjau konfigurasi instans Anda, lalu pilih Launch instans. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol.

Untuk mengaktifkan akses ke tag dalam metadata instance setelah peluncuran instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance Anda, lalu pilih Tindakan, Pengaturan instans, Izinkan tag dalam metadata instans.
4. Untuk mengizinkan akses ke tag dalam metadata instance, pilih kotak centang Izinkan.
5. Pilih Simpan.

AWS CLI

Untuk mengaktifkan akses ke tag dalam metadata instance selama peluncuran instance

Gunakan perintah run-instance dan tambahkan opsi berikut `--metadata-options`.

```
--metadata-options "InstanceMetadataTags=enabled"
```

Untuk mengaktifkan akses ke tag dalam metadata instance setelah peluncuran instance

Gunakan perintah modify-instance-metadata-options berikut.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567890abcdef0 \
    --instance-metadata-tags enabled
```

Untuk memverifikasi bahwa akses ke tag dalam metadata instance diaktifkan

Gunakan perintah describe-instance dan periksa nilainya. `InstanceMetadataTags`

```
aws ec2 describe-instances \
    --instance-ids i-1234567890abcdef0 \
    --query "Reservations[*].Instances[].MetadataOptions[].InstanceMetadataTags"
```

Berikut ini adalah output contoh. Nilainya adalah `enabled` atau `disabled`.

```
[
    "enabled"
]
```

PowerShell

Untuk mengaktifkan akses ke tag dalam metadata instance selama peluncuran instance

Gunakan New-EC2Instance cmdlet dan tambahkan parameter berikut-`MetadataOptions_InstanceMetadataTags`.

```
-MetadataOptions_InstanceMetadataTags enabled
```

Untuk mengaktifkan akses ke tag dalam metadata instance setelah peluncuran instance

Gunakan Edit-EC2InstanceMetadataOption cmdlet.

```
Edit-EC2InstanceMetadataOption `
```

```
    -InstanceId i-1234567890abcdef0 `
    -InstanceMetadataTags enabled
```

Untuk memverifikasi bahwa akses ke tag dalam metadata instance diaktifkan

Gunakan Get-EC2Instance cmdlet dan periksa nilainya. `InstanceMetadataTags`

```
(Get-EC2Instance `
    -
InstanceId i-1234567890abcdef0).Instances.MetadataOptions.InstanceMetadataTags.Value
```

Berikut ini adalah output contoh. Nilainya adalah `enabled` atau `disabled`.

```
enabled
```

## Mengambil tanda dari metadata instans

Setelah mengizinkan akses ke tag instance dalam metadata instans, Anda dapat mengakses `tags/instance` kategori dari metadata instans. Untuk informasi selengkapnya, lihat Mengakses metadata instans untuk instans EC2.

IMDSv2

Linux

Jalankan perintah berikut dari instance Linux Anda untuk mencantumkan semua kunci tag untuk instance tersebut.

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-
token-ttl-seconds: 21600"` \
    && curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/
meta-data/tags/instance
```

Contoh ini mendapatkan nilai kunci yang diperoleh dalam contoh sebelumnya. Permintaan IMDSv2 menggunakan token tersimpan yang dibuat menggunakan perintah pada contoh sebelumnya. Token tidak boleh kedaluwarsa.

```
curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/
tags/instance/tag-key
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda untuk mencantumkan semua kunci tag untuk instance.

```
$token = Invoke-RestMethod `
    -Headers @{"X-aws-ec2-metadata-token-ttl-seconds" = "21600"} `
    -Method PUT -Uri http://169.254.169.254/latest/api/token
```

```
Invoke-RestMethod `
    -Headers @{"X-aws-ec2-metadata-token" = $token} `
    -Method GET -Uri http://169.254.169.254/latest/meta-data/tags/instance
```

Contoh ini mendapatkan nilai kunci yang diperoleh dalam contoh sebelumnya. Permintaan IMDSv2 menggunakan token tersimpan yang dibuat menggunakan perintah pada contoh sebelumnya. Token tidak boleh kedaluwarsa.

```
Invoke-RestMethod `
    -Headers @{"X-aws-ec2-metadata-token" = $token} `
    -Method GET -Uri http://169.254.169.254/latest/meta-data/tags/instance/tag-key
```

IMDSv1

Linux

Jalankan perintah berikut dari instance Linux Anda untuk mencantumkan semua kunci tag untuk instance tersebut.

```
curl http://169.254.169.254/latest/meta-data/tags/instance
```

Contoh ini mendapatkan nilai kunci yang diperoleh dalam contoh sebelumnya.

```
curl http://169.254.169.254/latest/meta-data/tags/instance/tag-key
```

Windows

Jalankan cmdlet berikut dari instance Windows Anda untuk mencantumkan semua kunci tag untuk instance.

```
Invoke-RestMethod -Uri http://169.254.169.254/latest/meta-data/tags/instance
```

Contoh ini mendapatkan nilai kunci yang diperoleh dalam contoh sebelumnya.

```
Invoke-RestMethod -Uri http://169.254.169.254/latest/meta-data/tags/instance/tag-key
```

## Nonaktifkan akses ke tag dalam metadata instance

Anda dapat menonaktifkan akses ke tag instance dalam metadata instans. Anda tidak perlu menonaktifkan akses ke tag instance pada metadata instans saat diluncurkan karena dinonaktifkan secara default.

Console

Untuk menonaktifkan akses ke tag dalam metadata instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih sebuah instans, lalu pilih Tindakan, Pengaturan instans, Izinkan tanda dalam metadata instans.
4. Untuk menonaktifkan akses ke tag dalam metadata instance, kosongkan kotak centang Izinkan.
5. Pilih Simpan.

AWS CLI

Untuk menonaktifkan akses ke tag dalam metadata instance

Gunakan perintah modify- instance-metadata-options berikut.

```
aws ec2 modify-instance-metadata-options \
    --instance-id i-1234567890abcdef0 \
    --instance-metadata-tags disabled
```

PowerShell

Untuk menonaktifkan akses ke tag dalam metadata instance

Gunakan Edit-EC2InstanceMetadataOption cmdlet.

```
Edit-EC2InstanceMetadataOption `
    -InstanceId i-1234567890abcdef0 `
    -InstanceMetadataTag disabled
```

# Kuota layanan Amazon EC2

Saat Anda membuat Akun AWS, kami menetapkan kuota default (juga disebut sebagai batas) pada AWS sumber daya Anda per wilayah. Jika Anda mencoba melebihi kuota sumber daya, permintaan gagal. Misalnya, ada jumlah maksimum vCPU Amazon EC2 yang dapat Anda berikan untuk Instans di Wilayah. On-Demand Jika Anda mencoba meluncurkan instans di Wilayah dan permintaan ini akan menyebabkan penggunaan Anda melebihi kuota ini, permintaan gagal. Jika ini terjadi, Anda dapat mengurangi penggunaan sumber daya atau meminta peningkatan kuota.

Konsol Service Quotas adalah lokasi pusat tempat Anda dapat melihat dan mengelola kuota untuk AWS layanan, dan meminta peningkatan kuota untuk banyak sumber daya yang Anda gunakan. Gunakan kuota yang kami sediakan untuk mengelola AWS infrastruktur Anda. Rencanakan permintaan peningkatan kuota sebelum Anda membutuhkannya.

Dokumentasi kuota terkait

- Titik akhir dan kuota Amazon EC2
- Kuota jenis instans Amazon EC2
- Kuota untuk Amazon EBS
- Kuota Amazon VPC

## Melihat kuota Anda saat ini

Anda dapat melihat kuota untuk setiap Wilayah menggunakan konsol Kuota Layanan.

Untuk melihat kuota saat ini menggunakan konsol Kuota Layanan

1. Buka konsol Service Quotas di https://console.aws.amazon.com/servicequotas/home/services/ec2/quotas/.
2. Dari bilah navigasi (di bagian atas layar), pilih Wilayah.
3. Pada panel Kelola kuota, pilih layanan; misalnya, Amazon Elastic Compute Cloud (Amazon EC2). Pilih Lihat kuota.

4. Gunakan bidang filter untuk memfilter daftar berdasarkan nama sumber daya. Misalnya, masukkan **`On-Demand`** untuk menemukan kuota untuk On-Demand Instans.
5. Untuk melihat informasi selengkapnya, pilih nama kuota untuk membuka halaman detail kuota.

## Meminta peningkatan

Anda dapat meminta peningkatan kuota untuk setiap Wilayah.

Untuk meminta peningkatan menggunakan konsol Kuota Layanan

1. Buka konsol Service Quotas di https://console.aws.amazon.com/servicequotas/home/services/ec2/quotas/.
2. Dari bilah navigasi (di bagian atas layar), pilih Wilayah.
3. Gunakan bidang filter untuk memfilter daftar berdasarkan nama sumber daya. Misalnya, masukkan **`On-Demand`** untuk menemukan kuota untuk On-Demand Instans.
4. Jika kuota dapat disesuaikan, pilih kuota lalu pilih Minta peningkatan kuota.
5. Untuk Ubah nilai kuota, masukkan nilai kuota baru.
6. Pilih Minta.
7. Untuk melihat permintaan yang tertunda atau baru diselesaikan di konsol, pilih Dasbor dari panel navigasi. Untuk permintaan yang tertunda, pilih status permintaan untuk membuka penerimaan permintaan. Status awal dari permintaan adalah Tertunda. Setelah status berubah menjadi Kuota yang diminta, Anda akan melihat nomor kasus dengan Dukungan. Pilih nomor kasus untuk membuka tiket untuk permintaan Anda.

Untuk informasi selengkapnya, termasuk cara menggunakan AWS CLI atau SDK untuk meminta peningkatan kuota, lihat Meminta peningkatan kuota dalam Panduan Pengguna Service Quotas.

## Pembatasan pada email yang dikirim menggunakan port 25

Secara default, Amazon EC2 memungkinkan lalu lintas keluar melalui port 25 hanya ke alamat IPv4 pribadi. Lalu lintas melalui port 25 diblokir ke alamat IPv4 publik dan alamat IPv6.

Anda dapat meminta penghapusan pembatasan ini. Untuk informasi selengkapnya, lihat Bagaimana cara menghapus pembatasan pada port 25 dari instans Amazon EC2 atau fungsi Lambda saya?

Pembatasan ini tidak berlaku untuk lalu lintas keluar melalui port 25 yang ditujukan untuk:

- Alamat IP di blok CIDR utama VPC di mana antarmuka jaringan asal berada.
- Alamat IP dalam CIDR yang ditentukan dalam RFC 1918, RFC 6598, dan RFC 4193.

# Pantau sumber daya Amazon EC2

Pemantauan adalah bagian penting dalam menjaga keandalan, ketersediaan, dan kinerja instans Amazon EC2 dan AWS solusi Anda. Anda harus mengumpulkan data pemantauan dari semua bagian dalam AWS solusi Anda sehingga Anda dapat lebih mudah men-debug kegagalan multi-titik jika terjadi.

AWS menyediakan berbagai alat yang dapat Anda gunakan untuk memantau Amazon EC2. Amazon EC2 dan dasbor konsol CloudWatch memberikan pandangan sekilas tentang status lingkungan Amazon EC2 Anda. Selain itu, kami menyediakan yang berikut:

- Pemeriksaan status sistem — Pan AWS tau sistem yang diperlukan untuk menggunakan instance Anda untuk memastikan bahwa mereka berfungsi dengan baik. Pemeriksaan ini mendeteksi masalah dengan instance Anda yang memerlukan AWS keterlibatan untuk diperbaiki. Jika pemeriksaan status sistem gagal, Anda dapat memilih untuk menunggu AWS memperbaiki masalah tersebut atau Anda dapat memecahkannya sendiri (misalnya, dengan menghentikan dan memulai ulang atau mengakhiri dan mengganti instans). Contoh masalah yang menyebabkan kegagalan pemeriksaan status sistem meliputi:
  - Kehilangan konektivitas jaringan
  - Kehilangan daya sistem
  - Masalah perangkat lunak pada host fisik
  - Masalah perangkat keras pada host fisik yang memengaruhi jangkauan jaringan

  Untuk informasi selengkapnya, lihat Pemeriksaan status untuk instans Amazon EC2.
- Pemeriksaan status instans — Pantau perangkat lunak dan konfigurasi jaringan instans individual Anda. Pemeriksaan ini mendeteksi masalah yang memerlukan keterlibatan Anda untuk memperbaikinya. Jika pemeriksaan status instans gagal, biasanya Anda perlu menangani sendiri masalah tersebut (misalnya, dengan melakukan boot ulang instans tersebut atau membuat modifikasi dalam sistem operasi Anda). Contoh masalah yang mungkin menyebabkan kegagalan pemeriksaan status instans meliputi:
  - Pemeriksaan status sistem gagal
  - Konfigurasi jaringan atau pemulaian salah
  - Memori habis
  - Sistem file rusak
  - Kernel tidak kompatibel

Untuk informasi selengkapnya, lihat Pemeriksaan status untuk instans Amazon EC2.

- CloudWatch Alarm Amazon — Tonton satu metrik selama periode waktu yang Anda tentukan, dan lakukan satu atau lebih tindakan berdasarkan nilai metrik relatif terhadap ambang batas tertentu selama beberapa periode waktu. Tindakan ini adalah pengiriman notifikasi ke topik Amazon Simple Notification Service (Amazon SNS) atau kebijakan Amazon EC2 Auto Scaling. Alarm memanggil tindakan hanya untuk perubahan status berkelanjutan. CloudWatch alarm tidak akan memanggil tindakan hanya karena mereka berada dalam keadaan tertentu; keadaan harus telah berubah dan dipertahankan untuk sejumlah periode tertentu. Untuk informasi selengkapnya, lihat Memantau instans Anda menggunakan CloudWatch.
- Acara Amazon EventBridge — Otomatiskan AWS layanan Anda dan merespons secara otomatis peristiwa sistem. Peristiwa dari AWS layanan dikirimkan EventBridge hampir secara real time, dan Anda dapat menentukan tindakan otomatis yang akan dilakukan saat acara cocok dengan aturan yang Anda tulis. Untuk informasi selengkapnya, lihat the section called “Otomatisasi menggunakan EventBridge”.
- AWS CloudTrail log — Tangkap informasi terperinci tentang panggilan yang dilakukan ke Amazon EC2 API dan menyimpannya sebagai file log di Amazon S3. Anda dapat menggunakan CloudTrail log untuk menentukan panggilan mana yang dilakukan, alamat IP sumber untuk panggilan, siapa yang melakukan panggilan, dan kapan panggilan dilakukan. Untuk informasi selengkapnya, lihat the section called “Log panggilan API menggunakan CloudTrail”.
- CloudWatch agen — Kumpulkan log dan metrik tingkat sistem dari host dan tamu di instans EC2 dan server lokal Anda. Untuk informasi selengkapnya, lihat Mengumpulkan Metrik dan Log dari Instans dan On-Premises Server Amazon EC2 dengan CloudWatch Agen di Panduan CloudWatch Pengguna Amazon.

# Memantau status instans Amazon EC2 Anda

Anda dapat memantau status instans dengan melihat pemeriksaan status dan peristiwa terjadwal untuk instans Anda.

Pemeriksaan status memberi Anda informasi yang dihasilkan dari pemeriksaan otomatis yang dilakukan oleh Amazon EC2. Pemeriksaan otomatis ini mendeteksi apakah masalah tertentu memengaruhi instans Anda. Informasi pemeriksaan status, bersama dengan data yang disediakan oleh Amazon CloudWatch, memberi Anda visibilitas operasional terperinci ke setiap instans Anda.

Anda juga dapat melihat status acara tertentu yang dijadwalkan untuk instans Anda. Status peristiwa memberikan informasi tentang aktivitas mendatang yang direncanakan untuk instans Anda, seperti

boot ulang atau pemensiunan. Status tersebut juga memberikan informasi waktu mulai dan selesai terjadwal untuk setiap peristiwa.

Daftar Isi


- Pemeriksaan status untuk instans Amazon EC2
- Pemeriksaan status aplikasi
- Peristiwa perubahan status untuk instans Amazon EC2
- Acara terjadwal untuk instans Amazon EC2


## Pemeriksaan status untuk instans Amazon EC2

Dengan pemeriksaan status, Anda dapat dengan cepat menentukan apakah ada masalah yang dapat mencegah instans menjalankan aplikasi Anda. Amazon EC2 menyediakan empat jenis pemeriksaan status: sistem, instance, EBS terlampir, dan aplikasi. Pemeriksaan status sistem, instans, dan EBS terlampir dikelola oleh Amazon EC2 dan dijalankan secara otomatis di setiap instans. Pemeriksaan status aplikasi bersifat opt-in dan memantau respons HTTP atau HTTPS dari aplikasi yang berjalan di instans Anda. Anda dapat melihat hasil semua pemeriksaan status di konsol Amazon EC2 AWS CLI, atau AWS SDK.

Pemeriksaan status dilakukan setiap menit dan menghasilkan status lulus atau gagal. Jika semua pemeriksaan lulus, status keseluruhan instans adalah OK. Jika satu atau beberapa pemeriksaan gagal, status keseluruhannya adalah terganggu. Pemeriksaan status sistem, instans, dan EBS terlampir dikelola oleh Amazon EC2 dan tidak dapat dinonaktifkan atau dihapus. Pemeriksaan status aplikasi bersifat opt-in; Anda membuat, mengaitkan, dan mengelolanya sendiri.

Ketika pemeriksaan status gagal, CloudWatch metrik yang sesuai untuk pemeriksaan status bertambah. Untuk informasi selengkapnya, lihat Metrik pemeriksaan status. Anda dapat menggunakan metrik ini untuk membuat alarm CloudWatch yang dipicu berdasarkan hasil pemeriksaan status. Misalnya, Anda dapat membuat alarm untuk memperingatkan Anda jika pemeriksaan status gagal pada instans tertentu. Untuk informasi selengkapnya, lihat Buat CloudWatch alarm untuk instans Amazon EC2 yang gagal dalam pemeriksaan status.

Anda juga dapat membuat CloudWatch alarm Amazon yang memantau instans Amazon EC2 dan secara otomatis memulihkan instans jika menjadi rusak karena masalah yang mendasarinya. Untuk informasi selengkapnya, lihat Pemulihan instans otomatis.

Daftar Isi

- Tipe pemeriksaan status
- Melihat pemeriksaan status untuk instans Amazon EC2
- Buat CloudWatch alarm untuk instans Amazon EC2 yang gagal dalam pemeriksaan status


## Tipe pemeriksaan status

Ada empat jenis pemeriksaan status.


- Pemeriksaan status sistem
- Pemeriksaan status instans
- Pemeriksaan status EBS terlampir
- Pemeriksaan status aplikasi


### Pemeriksaan status sistem

Pemeriksaan status AWS sistem memantau sistem tempat instance Anda berjalan. Pemeriksaan ini mendeteksi masalah yang mendasari instans, yang memerlukan keterlibatan AWS untuk diperbaiki. Ketika pemeriksaan status sistem gagal, Anda dapat memilih untuk menunggu AWS untuk memperbaiki masalah, atau Anda dapat menyelesaikannya sendiri. Untuk instans yang didukung oleh Amazon EBS, Anda dapat menghentikan dan memulai instans sendiri, yang pada sebagian besar kasus akan membuat instans dimigrasikan ke host baru. Untuk instans yang didukung oleh penyimpanan instance (hanya didukung untuk instans Linux), Anda dapat menghentikan dan mengganti instance. Perhatikan bahwa volume penyimpanan instans bersifat sementara dan semua data akan hilang saat instans dihentikan.

Berikut adalah contoh masalah yang dapat menyebabkan kegagalan pemeriksaan status sistem:

- Hilangnya konektivitas jaringan
- Kehilangan daya sistem
- Masalah perangkat lunak pada host fisik
- Masalah perangkat keras pada hosting fisik yang memengaruhi jangkauan jaringan

Jika pemeriksaan status sistem gagal, kami menambah metrik StatusCheckFailed_System.

Instans bare metal

Jika Anda melakukan restart pada instans bare metal, pemeriksaan status sistem mungkin untuk sementara mengembalikan status gagal. Ketika instans tersedia, pemeriksaan status sistem seharusnya kembali ke status lulus.

Pemeriksaan status instans

Pemeriksaan status instans memantau perangkat lunak dan konektivitas jaringan dari masing-masing instance Anda. Amazon EC2 memeriksa kondisi instans dengan mengirimkan permintaan protokol resolusi alamat (ARP) ke antarmuka jaringan (NIC). Pemeriksaan ini mendeteksi masalah yang memerlukan keterlibatan Anda untuk memperbaikinya. Jika pemeriksaan status instans gagal, Anda biasanya harus mengatasi sendiri masalah tersebut (misalnya, dengan melakukan boot ulang instans atau membuat perubahan konfigurasi instans).

> **Note**
>
> Distribusi Linux terbaru yang digunakan `systemd-networkd` untuk konfigurasi jaringan mungkin melaporkan pemeriksaan kesehatan secara berbeda dari distribusi sebelumnya. Selama proses boot, jenis jaringan ini dapat dimulai lebih awal dan berpotensi selesai sebelum tugas startup lain yang juga dapat memengaruhi kesehatan instans. Pemeriksaan status yang bergantung pada ketersediaan jaringan dapat melaporkan status sehat sebelum tugas lain selesai.

Berikut adalah contoh masalah yang dapat menyebabkan kegagalan pemeriksaan status instans:

- Pemeriksaan status sistem gagal
- Konfigurasi jaringan atau pemulaian salah
- Memori habis
- Sistem file rusak
- Kernel tidak kompatibel
- Selama reboot, pemeriksaan status instance dapat melaporkan kegagalan hingga instans tersedia lagi.

Jika pemeriksaan status instance gagal, kami menambah metrik _StatusCheckFailed _Instance.

Instans bare metal

Jika Anda melakukan restart pada instans bare metal, pemeriksaan status instance mungkin untuk sementara mengembalikan status gagal. Ketika instans tersedia, pemeriksaan status instans seharusnya kembali ke status lulus.

Pemeriksaan status EBS terlampir

Pemeriksaan status EBS terlampir memantau apakah volume Amazon EBS yang dilampirkan ke instans dapat dijangkau dan dapat menyelesaikan operasi. I/O `StatusCheckFailed_AttachedEBS`Metrik adalah nilai biner yang menunjukkan penurunan nilai jika satu atau lebih volume EBS yang dilampirkan ke instans tidak dapat menyelesaikan I/O operasi. Pemeriksaan status ini mendeteksi masalah yang mendasari komputasi atau infrastruktur Amazon EBS. Ketika metrik pemeriksaan status EBS yang dilampirkan gagal, Anda dapat menunggu AWS untuk menyelesaikan masalah, atau Anda dapat mengambil tindakan, seperti mengganti volume yang terpengaruh atau menghentikan dan memulai ulang instance.

Berikut adalah contoh masalah yang dapat menyebabkan kegagalan pemeriksaan status EBS terlampir:

- Masalah perangkat keras atau perangkat lunak pada subsistem penyimpanan yang mendasari volume EBS
- Masalah perangkat keras pada host fisik yang memengaruhi jangkauan volume EBS
- Masalah konektivitas antara instans dan volume EBS

Anda dapat menggunakan metrik `StatusCheckFailed_AttachedEBS` untuk membantu meningkatkan ketahanan beban kerja Anda. Anda dapat menggunakan metrik ini untuk membuat CloudWatch alarm Amazon yang dipicu berdasarkan hasil pemeriksaan status. Misalnya, Anda dapat melakukan failover ke instans sekunder atau Zona Ketersediaan saat mendeteksi adanya dampak yang berkepanjangan. Atau, Anda dapat memantau I/O kinerja setiap volume yang dilampirkan menggunakan CloudWatch metrik EBS untuk mendeteksi dan mengganti volume yang rusak. Jika pemeriksaan status EBS menunjukkan adanya gangguan dan beban kerja Anda tidak mengarah I/O ke volume EBS terlampir, hentikan dan mulai instans untuk memindahkannya ke host baru. Ini dapat menyelesaikan masalah host mendasar yang berdampak pada jangkauan volume EBS. Untuk informasi selengkapnya, lihat CloudWatch Metrik Amazon untuk Amazon EBS.

Anda juga dapat mengonfigurasi grup Amazon EC2 Auto Scaling untuk mendeteksi kegagalan pemeriksaan status EBS yang terlampir, lalu mengganti instans yang terpengaruh dengan yang baru. Untuk informasi selengkapnya, lihat Mem antau dan mengganti instans Auto Scaling dengan volume Amazon EBS yang rusak di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

> **Note**
>
> Metrik pemeriksaan status EBS yang terlampir hanya tersedia untuk instans Nitro.

Pemeriksaan status aplikasi

Gunakan pemeriksaan status aplikasi untuk memantau jangkauan jaringan dan ketersediaan aplikasi yang berjalan di instans Amazon EC2 Anda. Dengan pemeriksaan status aplikasi, Anda dapat memantau respons HTTP dan HTTPS di tingkat aplikasi. Pemeriksaan status aplikasi hanya berjalan pada instans yang Anda kaitkan dengannya.

Anda mengonfigurasi setiap pemeriksaan status aplikasi untuk meminta jalur HTTP pada titik akhir aplikasi Anda dan menentukan kode respons yang menunjukkan respons yang sehat.

Anda dapat menekan pemeriksaan status aplikasi selama restart atau penerapan aplikasi untuk menghindari kegagalan positif palsu selama waktu henti yang diharapkan. Untuk informasi selengkapnya, lihat Menangani penerapan, tambalan di tempat, dan penggantian.

Berikut ini adalah contoh masalah yang dapat menyebabkan pemeriksaan status aplikasi gagal:

- Proses aplikasi macet atau berhenti merespons
- Aplikasi mengembalikan kode respons HTTP yang menunjukkan kesalahan layanan aplikasi internal atau masalah lainnya
- Port aplikasi tidak dapat dijangkau
- Masalah perangkat lunak atau perangkat keras yang mendasarinya telah menyebabkan instans Anda gagal dan menjadi tidak responsif

Jika pemeriksaan status aplikasi gagal, kami menambah met `StatusCheckFailed_Application` rik.

Untuk informasi tentang mengkonfigurasi pemeriksaan status aplikasi, lihatPemeriksaan status aplikasi.

## Melihat pemeriksaan status untuk instans Amazon EC2

Jika instans Anda memiliki pemeriksaan status yang gagal, Anda biasanya harus mengatasi sendiri masalah tersebut (misalnya, dengan melakukan boot ulang instans atau membuat perubahan konfigurasi instans). Untuk memecahkan sendiri masalah kegagalan pemeriksaan status sistem atau

instans, lihat Memecahkan masalah instans Amazon EC2 Linux dengan pemeriksaan status yang gagal.

Console

Untuk melihat pemeriksaan status

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pada halaman Instans, kolom Pemeriksaan status menampilkan status operasional setiap instans.
4. Untuk melihat status instans tertentu, pilih instans, lalu pilih tab Status dan alarm.
5. Untuk meninjau CloudWatch metrik untuk pemeriksaan status, pada tab Status dan alarm, perluas Metrik untuk melihat grafik untuk metrik berikut:
   - Pemeriksaan status sistem gagal
   - Pemeriksaan status instans gagal
   - Pemeriksaan status gagal untuk EBS terlampir
   - Pemeriksaan status gagal untuk aplikasi

   Untuk informasi selengkapnya, lihat the section called “Metrik pemeriksaan status”.

AWS CLI

Untuk melihat pemeriksaan status

Gunakan perintah describe-instance-status.

Contoh: Dapatkan status semua instans yang sedang berjalan

```
aws ec2 describe-instance-status
```

Contoh: Dapatkan status semua instance

```
aws ec2 describe-instance-status --include-all-instances
```

Contoh: Mendapatkan status satu instance yang sedang berjalan

```
aws ec2 describe-instance-status --instance-ids i-1234567890abcdef0
```

Contoh: Dapatkan semua instance dengan status `impaired`

```
aws ec2 describe-instance-status \
--filters Name=instance-status.status,Values=impaired
```

PowerShell

Untuk melihat pemeriksaan status

Gunakan perintah Get-EC2InstanceStatus.

Contoh: Dapatkan status semua instans yang sedang berjalan

```
Get-EC2InstanceStatus
```

Contoh: Dapatkan status semua instance

```
Get-EC2InstanceStatus -IncludeAllInstance $true
```

Contoh: Mendapatkan status satu instance yang sedang berjalan

```
Get-EC2InstanceStatus -InstanceId i-1234567890abcdef0
```

Contoh: Dapatkan semua instance dengan status `impaired`

```
Get-EC2InstanceStatus \
-Filter @{Name="instance-status.status"; Values="impaired"}
```

## Buat CloudWatch alarm untuk instans Amazon EC2 yang gagal dalam pemeriksaan status

Anda dapat menggunakan metrik pemeriksaan status untuk membuat CloudWatch alarm untuk memberi tahu Anda saat instans mengalami pemeriksaan status yang gagal.

Pemeriksaan status dan alarm pemeriksaan status dapat sementara memasuki status data yang tidak mencukupi jika ada titik data metrik yang hilang. Meskipun jarang, ini dapat terjadi ketika ada

gangguan dalam sistem pelaporan metrik, bahkan ketika instans sehat. Kami menyarankan Anda memperlakukan status ini sebagai data yang hilang alih-alih kegagalan pemeriksaan status atau pelanggaran alarm. Ini sangat penting ketika mengambil tindakan berhenti, menghentikan, me-reboot, atau memulihkan instans sebagai respons.

Console

Contoh ini mengonfigurasi alarm yang mengirimkan pemberitahuan ketika instance gagal dalam pemeriksaan status. Anda dapat menghentikan, menghentikan, atau memulihkan instance secara opsional.

Untuk membuat alarm pemeriksaan status

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, pilih tab Pemeriksaan Status, dan pilih Tindakan, Buat alarm pemeriksaan status.
4. Pada halaman Kelola CloudWatch alarm, di bawah Tambah atau edit alarm, pilih Buat alarm.
5. Untuk Notifikasi alarm, aktifkan sakelar guna mengonfigurasi notifikasi Amazon Simple Notification Service (Amazon SNS). Pilih topik Amazon SNS yang ada atau masukkan nama untuk membuat topik baru.

   Jika Anda menambahkan alamat email ke daftar penerima atau membuat topik baru, Amazon SNS mengirimkan email konfirmasi ke setiap alamat baru. Setiap penerima harus memilih tautan konfirmasi di email. Hanya alamat yang dikonfirmasi yang menerima pemberitahuan peringatan.
6. Untuk Tindakan alarm, aktifkan tombol untuk menentukan tindakan yang perlu dilakukan saat alarm dipicu. Pilih tindakan.
7. Untuk Ambang batas alarm, pilih metrik dan kriteria alarm.

   Anda dapat membiarkan pengaturan tetap default untuk Kelompokkan sampel berdasarkan (Rata-rata) dan Tipe data untuk sampel (Pemeriksaan status failed:either), atau Anda dapat mengubah pengaturan tersebut sesuai dengan kebutuhan.

   Untuk Periode berturut-turut, atur jumlah periode yang ingin Anda evaluasi dan, pada Periode, masukkan durasi periode evaluasi sebelum memicu alarm dan mengirimkan email.
8. (Opsional) Untuk Data metrik sampel, pilih Tambahkan ke dasbor.

9. Pilih Buat.

Jika Anda perlu mengubah alarm status instance, Anda dapat mengeditnya.

Untuk mengedit alarm pemeriksaan status

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans dan pilih Tindakan, Pemantauan, Kelola CloudWatch alarm.
4. Pada halaman Kelola CloudWatch alarm, di bawah Tambah atau edit alarm, pilih Edit alarm.
5. Untuk Cari alarm, pilih alarm.
6. Setelah Anda selesai membuat perubahan, pilih Perbarui.

AWS CLI

Dalam contoh berikut, alarm menerbitkan pemberitahuan ke topik SNS ketika instance gagal dalam pemeriksaan instans atau pemeriksaan status sistem setidaknya selama dua periode berturut-turut. Met CloudWatch rik yang digunakan adalah`StatusCheckFailed`.

Untuk membuat alarm pemeriksaan status

1. Pilih topik SNS yang ada atau buat baru. Untuk informasi selengkapnya, lihat Mengakses Amazon SNS di AWS CLI dalam Panduan AWS Command Line Interface Pengguna.
2. Gunakan perintah list-metrics berikut untuk melihat metrik Amazon CloudWatch yang tersedia untuk Amazon EC2.

```
aws cloudwatch list-metrics --namespace AWS/EC2
```

3. Gunakan perintah put-metric-alarm berikut untuk membuat alarm.

```
aws cloudwatch put-metric-alarm \
    --alarm-name StatusCheckFailed-Alarm-for-i-1234567890abcdef0 \
    --metric-name StatusCheckFailed \
    --namespace AWS/EC2 \
    --statistic Maximum \
    --dimensions Name=InstanceId,Value=i-1234567890abcdef0 \
    --unit Count \
```

```
    --period 300 \
    --evaluation-periods 2 \
    --threshold 1 \
    --comparison-operator GreaterThanOrEqualToThreshold \
    --alarm-actions arn:aws:sns:us-west-2:111122223333:my-sns-topic
```

Periode adalah kerangka waktu, dalam detik, saat metrik Amazon CloudWatch dikumpulkan. Contoh ini menggunakan 300, yaitu 60 detik dikalikan 5 menit. Periode evaluasi adalah jumlah periode berturut-turut yang nilai metriknya harus dibandingkan dengan ambang batas. Contoh ini menggunakan 2. Tindakan alarm adalah tindakan yang harus dilakukan saat alarm ini dipicu.

PowerShell

Untuk membuat alarm pemeriksaan status

Gunakan Write-CWMetricAlarm cmdlet sebagai berikut untuk menerbitkan notifikasi ke topik SNS ketika instance gagal memeriksa status setidaknya selama dua periode berturut-turut.

```
Write-CWMetricAlarm `
    -AlarmName "StatusCheckFailed-Alarm-for-i-1234567890abcdef0" `
    -MetricName "StatusCheckFailed" `
    -Namespace "AWS/EC2" `
    -Statistic "Maximum" `
    -Dimension @{Name="InstanceId"; Values="i-1234567890abcdef0"} `
    -Unit "Count" `
    -Period 300 `
    -EvaluationPeriod 2 `
    -Threshold 1 `
    -ComparisonOperator "GreaterThanOrEqualToThreshold" `
    -AlarmAction "arn:aws:sns:us-west-2:111122223333:my-sns-topic"
```

Periode adalah kerangka waktu, dalam detik, saat metrik Amazon CloudWatch dikumpulkan. Contoh ini menggunakan 300, yaitu 60 detik dikalikan 5 menit. Periode evaluasi adalah jumlah periode berturut-turut yang nilai metriknya harus dibandingkan dengan ambang batas. Contoh ini menggunakan 2. Tindakan alarm adalah tindakan yang harus dilakukan saat alarm ini dipicu.

# Pemeriksaan status aplikasi

Pemeriksaan status aplikasi membantu Anda memantau kinerja dan kesehatan aplikasi yang berjalan di Amazon EC2. Dengan pemeriksaan status aplikasi, Anda dapat mendeteksi dan menanggapi gangguan kesehatan aplikasi dengan memantau aplikasi Anda melalui jalur dan port yang dapat dikonfigurasi. Misalnya, Anda dapat menggunakan pemeriksaan status aplikasi untuk mengonfirmasi bahwa server web Anda mendengarkan port yang diharapkan dan menerima koneksi baru.

Pemeriksaan status aplikasi memantau respons HTTP dan HTTPS aplikasi Anda di jalur dan port yang dapat dikonfigurasi. Mereka berjalan setiap 60 detik dan terintegrasi dengan Amazon EC2 Auto Scaling, sehingga Anda dapat mengotomatiskan penggantian instans yang aplikasinya rusak.

Daftar Isi


- Cara kerja pemeriksaan status aplikasi
- Memulai pemeriksaan status aplikasi
- Opsi konfigurasi
- Pengaturan default
- Integrasi Penskalaan Otomatis Amazon EC2
- Menangani penerapan, tambalan di tempat, dan penggantian
- Menguji pemeriksaan status aplikasi baru
- Jaringan lanjutan
- Praktik terbaik
- Pemecahan masalah
- Pantau pemeriksaan status aplikasi
- Keamanan dan izin
- Harga
- Kuota


## Cara kerja pemeriksaan status aplikasi

Pemeriksaan status aplikasi mengirim permintaan HTTP atau HTTPS ke titik akhir yang mendengarkan di port jaringan pada instans Anda setiap 60 detik. AWS membandingkan kode respons dengan pencocokan kode status yang Anda konfigurasikan. Cek ditandai terganggu setelah

sejumlah permintaan gagal berturut-turut, dan sehat kembali setelah sejumlah permintaan berhasil berturut-turut. Keduanya dihitung secara default ke 2 dan dapat dikonfigurasi. Untuk informasi selengkapnya, lihat Ambang evaluasi.

> **Note**
>
> Pemeriksaan status aplikasi mengirimkan permintaan pemeriksaan kesehatan HTTP/2. Pemeriksaan protokol HTTPS tidak memvalidasi sertifikat server.

Selama reboot, pemeriksaan status aplikasi melaporkan kegagalan hingga instans tersedia lagi karena aplikasi tidak dapat menanggapi permintaan pemeriksaan kesehatan saat sistem operasi memulai ulang.

Arsitektur jaringan

Pemeriksaan status aplikasi berasal dari layanan pemeriksaan status aplikasi Amazon EC2. Untuk menjangkau instans Anda, AWS buat antarmuka jaringan elastis terkelola (ENI) di VPC Anda. AWS membuat satu ENI per kombinasi subnet sumber dan grup keamanan yang memiliki instans terkait. AWS membuat ENI terkelola ketika pemeriksaan status aplikasi terlebih dahulu memerlukan kombinasi itu, dan menghapusnya ketika tidak ada pemeriksaan status aplikasi yang tersisa yang memerlukannya. ENI terkelola tidak dihitung terhadap batas ENI instans Anda, tetapi dihitung terhadap ku ota Antarmuka Jaringan per Wilayah untuk akun Anda, yang diberlakukan per Zona Ketersediaan. Untuk informasi selengkapnya, lihat kuota Amazon VPC.

Secara default, Amazon EC2 menyembunyikan antarmuka jaringan terkelola ini dari operasi daftar konsol dan API untuk akun yang tidak memiliki sumber daya terkelola sebelum pengaturan ini tersedia. Untuk mengubah visibilitasnya, lihat Pengaturan visibilitas sumber daya terkelola.

AWS membuat satu antarmuka jaringan terkelola untuk setiap kombinasi subnet sumber dan grup keamanan di antara instans terkait Anda. Jumlah antarmuka terkelola bertambah seiring dengan banyaknya kombinasi subnet dan grup keamanan yang berbeda yang digunakan instans yang dipantau. Mengkonsolidasikan instans yang dipantau ke dalam kombinasi subnet dan grup keamanan yang lebih sedikit mengurangi jumlah antarmuka yang dikelola. Misalnya, 200 instans yang tersebar di 2 subnet yang semuanya menggunakan satu grup keamanan menghasilkan 2 antarmuka terkelola. 200 instans yang sama menggunakan 3 grup keamanan di 2 subnet tersebut menghasilkan hingga 6 antarmuka terkelola. Setiap antarmuka sesuai dengan satu subnet dan kombinasi grup keamanan.

Pemeriksaan status aplikasi mencapai instans Anda dari sudut pandang pribadi dalam VPC Anda. Cakupan menjelaskan dari mana cek berasal, bukan properti alamat IP instance Anda. AWS membuat ENI terkelola di subnet dalam VPC Anda dan mencapai instance melalui jalur jaringan pribadi.

Dengan jalur AWS jaringan terkelola, lalu lintas pemeriksaan kesehatan berasal dari AWS instans Amazon EC2 terkelola di Zona Ketersediaan yang sama dengan instans target (atau Zona Ketersediaan induk untuk target Zona Lokal). Lalu lintas berjalan melalui jaringan AWS internal dan tidak melintasi internet publik. Untuk informasi selengkapnya, lihat FAQ Amazon VPC di situs web Amazon Web Services.

Dengan jalur jaringan yang dikelola pelanggan, Anda memilih subnet sumber, sehingga Anda dapat menjalankan pemeriksaan dari Zona Ketersediaan yang berbeda dari target. Untuk informasi selengkapnya, lihat the section called “Cross-Availability Pemantauan zona”.

AWS jalur jaringan yang dikelola dan dikelola pelanggan

Pemeriksaan status aplikasi mendukung dua mode orientasi yang menentukan siapa yang memilih subnet sumber dan grup keamanan untuk pemeriksaan kesehatan ENI dan subnet tujuan dan grup keamanan untuk instans target.

AWS jalur jaringan yang dikelola

AWS memilih subnet sumber dan grup keamanan untuk pemeriksaan kesehatan ENI dan subnet tujuan dan grup keamanan untuk instans target.

Customer-managed jalur jaringan

Anda menentukan subnet sumber dan grup keamanan untuk pemeriksaan kesehatan ENI dan subnet tujuan dan grup keamanan untuk instans target.

Gunakan jalur jaringan yang dikelola pelanggan saat Anda perlu mengontrol lalu lintas pemeriksaan kesehatan subnet dan grup keamanan mana yang berasal, seperti ketika VPC Anda memiliki segmentasi jaringan yang ketat, aturan firewall, atau persyaratan kepatuhan yang membatasi sumber mana yang dapat mencapai titik akhir aplikasi Anda.

Anda memilih mode dengan memasukkan atau menghilangkan `--health-check-paths` parameter dalam perintah create. Jika Anda menghilangkan `--health-check-paths` parameter, pilih subnet AWS sumber dan tujuan dan grup keamanan (jalur jaringan terkel AWS ola). Jika Anda menyertakan `--health-check-paths` parameter, Anda mengelolanya (jalur jaringan yang dikelola pelanggan).

### Versi IP

Setiap pemeriksaan status aplikasi dikaitkan dengan versi IP tunggal (IPv4 atau IPv6). Untuk memantau instance melalui IPv4 dan IPv6, buat dua pemeriksaan status aplikasi terpisah dan kaitkan keduanya dengan instance.

Pemeriksaan mencapai instans Anda dari dalam VPC untuk IPv4 dan IPv6.

### Periksa nilai status

Setiap pemeriksaan individu melaporkan salah satu status berikut:

- `passed`: cek berhasil diselesaikan
- `failed`: pemeriksaan gagal. Respons mencakup kode status HTTP yang dikembalikan oleh aplikasi Anda. Untuk panduan interpretasi dan remediasi, lihatPemecahan masalah.
- `initializing`: cek belum menyelesaikan evaluasi pertamanya
- `insufficient-data`: cek tidak menerima data yang cukup untuk menentukan hasil
- `not-applicable`: cek tidak terkait dengan instance

Status aplikasi keseluruhan yang dilaporkan untuk instans menggabungkan semua hasil pemeriksaan individu. Status keseluruhan adalah salah satu dari berikut ini:

- `ok`: semua cek lulus
- `impaired`: satu atau lebih pemeriksaan gagal
- `initializing`: satu atau lebih pemeriksaan belum menyelesaikan evaluasi pertama mereka
- `insufficient-data`: satu atau lebih cek melaporkan data yang tidak mencukupi
- `not-applicable`: semua pemeriksaan status aplikasi terkait dikecualikan dari agregasi
- `suppressed`: evaluasi pemeriksaan status aplikasi ditekan untuk instance

### Agregasi

Anda dapat menandai setiap pemeriksaan status aplikasi sebagai termasuk dalam atau dikecualikan dari status keseluruhan instans. Secara default, cek adalah`included`.

`included`

Pemeriksaan berkontribusi pada status keseluruhan untuk instance dan Amazon EC2 Auto Scaling menggunakannya.

`excluded`

Pemeriksaan melaporkan status individualnya tetapi tidak berkontribusi pada status keseluruhan untuk instans dan Amazon EC2 Auto Scaling tidak menggunakannya. Gunakan pengaturan ini untuk memvalidasi pemeriksaan baru dalam produksi tanpa mempengaruhi status keseluruhan atau memicu penggantian Amazon EC2 Auto Scaling. Ini adalah alur kerja yang disarankan saat menambahkan cek ke beban kerja produksi yang ada; lihatMenguji pemeriksaan status aplikasi baru.

## Memulai pemeriksaan status aplikasi

### Prasyarat

Sebelum Anda membuat pemeriksaan status aplikasi, pastikan Anda memiliki yang berikut:

- VPC dengan instans yang ingin Anda pantau.
- Titik akhir aplikasi pada setiap instance yang dapat merespons permintaan HTTP atau HTTPS pada port dan jalur HTTP yang akan Anda konfigurasikan.
- Grup keamanan pada setiap instance tujuan yang memungkinkan lalu lintas masuk pada port cek dari grup keamanan sumber yang digunakan oleh pemeriksaan status aplikasi. Lihat Keamanan dan izin.

### Langkah 1: Konfigurasikan aplikasi Anda

Konfigurasikan titik akhir aplikasi Anda untuk menanggapi permintaan HTTP atau HTTPS pada port dan jalur HTTP yang akan Anda tentukan saat membuat pemeriksaan. Kembalikan kode respons yang disertakan dalam pencocokan kode status Anda untuk menunjukkan aplikasi sehat.

Pastikan grup keamanan instans tujuan mengizinkan lalu lintas masuk pada port cek dari grup keamanan sumber yang digunakan oleh pemeriksaan status aplikasi. Untuk jalur jaringan terkelola, AWS berikan grup keamanan sumber pada pembuatan cek. Untuk jalur jaringan yang dikelola pelanggan, Anda menentukan grup keamanan sumber saat membuat cek.

### Langkah 2: Buat definisi cek

Gunakan AWS CLI untuk membuat pemeriksaan status aplikasi.

Console

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, di bawah Instans, pilih Pemer iksaan status aplikasi.
3. Pilih Buat pemeriksaan status aplikasi.
4. Di bawah logika pemeriksaan kesehatan, konfigurasikan yang berikut ini:
   - Protokol: pilih HTTP atau HTTPS.
   - Port: masukkan port tempat aplikasi Anda mendengarkan.
   - Path (opsional): masukkan jalur HTTP untuk meminta, misalnya `/healthcheck`.
   - Versi IP: pilih IPv4 atau IP v6.
   - Indeks perangkat: indeks perangkat antarmuka jaringan untuk diperiksa. Default-nya adalah `0`.
5. Di bawah Kontrol dan ambang batas, atur batas waktu, dan secara opsional pencocokan kode status, ambang kegagalan, ambang batas keberhasilan , dan masa tenggang inisialisasi. Interval pemeriksaan ditetapkan pada 60 detik.
6. Di bawah Agregasi, pilih Ter masuk untuk pemeriksaan untuk berkontribusi pada status aplikasi secara keseluruhan dan mendorong Penskalaan Otomatis Amazon EC2, atau Dikecualikan untuk melaporkan pemeriksaan tanpa mempengaruhi status keseluruhan.
7. Di bawah Jalur pemeriksaan kesehatan, simpan Jangan tentukan jalur jaringan (recommended/default) agar Amazon EC2 menempatkan antarmuka jaringan pemeriksaan kesehatan di subnet instans Anda, atau pilih T entukan jalur jaringan (lanjutan) untuk menentukan sendiri subnet sumber, grup keamanan, dan tujuan.
8. (Opsional) Tambahkan tag Nama dan Tag lainnya.
9. Pilih Buat pemeriksaan status aplikasi.

AWS CLI

Untuk menggunakan AWS jalur jaringan terkelola, hilangkan `--health-check-paths` parameter dan biarkan AWS pilih subnet sumber dan tujuan serta grup keamanan.

```
aws ec2 create-application-status-check \
        --protocol https \
        --port 443 \
        --path "/health" \
```

```
        --status-code-matcher "200"
```

Untuk menggunakan jalur jaringan yang dikelola pelanggan, sertakan parameter. `--health-check-paths` Setiap jalur pemeriksaan kesehatan berisi sumber (subnet dan grup keamanan untuk pemeriksaan kesehatan ENI) dan satu atau lebih tujuan (subnet dan grup keamanan untuk instans target).

```
aws ec2 create-application-status-check \
        --protocol https \
        --port 443 \
        --path "/health" \
        --status-code-matcher "200" \
        --health-check-paths '[{"Source":
{"SubnetId":"subnet-111","SecurityGroupId":"sg-aaa"},"Destinations":
[{"SubnetId":"subnet-222","SecurityGroupId":"sg-bbb"}]}]'
```

Langkah 3: Mengaitkan cek dengan instance

Kaitkan pemeriksaan dengan instans yang ingin Anda pantau, baik berdasarkan ID instance atau dengan tag.

Console

1. Di panel navigasi, di bawah  Instans, pilih  Pemeriksaan status aplikasi, dan pilih centang.
2. Pilih  Kelola asosiasi pemeriksaan status, lalu pilih  Kelola asosiasi berdasarkan ID sumber daya  atau  Kelola asosiasi berdasarkan tag.
3. Untuk mengaitkan dengan semua instance dalam grup Penskalaan Otomatis, pilih  Kelola asosiasi berdasarkan tag  dan masukkan `aws:autoscaling:groupName` sebagai kunci tag dan nama grup Penskalaan Otomatis Anda sebagai nilainya.
4. Pilih Kaitkan.

AWS CLI

Dengan ID instance:

```
aws ec2 associate-application-status-check \
        --application-status-check-id asc-1234567890abcdef0 \
        --instance-ids i-0123456789abcdef0
```

Dengan tag:

```
aws ec2 associate-application-status-check \
        --application-status-check-id asc-1234567890abcdef0 \
        --target-tag-associations Key=Environment,Value=production
```

Untuk mengaitkan dengan semua instance dalam grup Auto Scaling, gunakan tag `aws:autoscaling:groupName` sistem:

```
aws ec2 associate-application-status-check \
        --application-status-check-id asc-1234567890abcdef0 \
        --target-tag-associations Key=aws:autoscaling:groupName,Value=my-asg
```

Operasi asosiasi dan disasosiasi mengembalikan hasil keberhasilan dan kegagalan per instans. Jika beberapa contoh tidak dapat dikaitkan (misalnya, karena pemeriksaan sudah dikaitkan), instance tersebut muncul dalam hasil yang tidak berhasil dengan alasan.

Langkah 4: Lihat hasil

Melihat status kesehatan aplikasi per instans.

Console

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans, lalu pilih tab Status dan alarm.
4. Di bawah Pemeriksaan status aplikasi, tinjau status keseluruhan dan status individu setiap cek terkait.

AWS CLI

```
aws ec2 describe-application-status \
        --instance-ids i-0123456789abcdef0
```

Respons mencakup status aplikasi secara keseluruhan dan, untuk setiap pemeriksaan terkait, status pemeriksaan dan, untuk pemeriksaan gagal, kode status HTTP yang dikembalikan oleh aplikasi Anda.

Contoh respons:

```
{
        "ApplicationStatuses": [
            {
                "InstanceId": "i-0123456789abcdef0",
                "ApplicationStatus": {
                    "Status": "ok",
                    "Details": [
                        {
                            "ApplicationStatusCheckId": "asc-1234567890abcdef0",
                            "Status": "passed",
                            "Reason": {
                                "Code": "ResponseCodeMatched",
                                "StatusCode": 200,
                                "Protocol": "HTTP"
                            }
                        }
                    ]
                }
            }
        ]
    }
```

Untuk melihat definisi cek (bukan status per instans), gunakan descri be-application-status-checks. Perintah ini mengembalikan konfigurasi pemeriksaan status aplikasi Anda, termasuk protokol, port, jalur HTTP, dan pengaturan pencocokan kode status.

## Opsi konfigurasi

Pemeriksaan status aplikasi menerima beberapa parameter konfigurasi. Bagian ini menjelaskan parameter di mana perilaku tidak terbukti dengan sendirinya dari nama parameter. Untuk daftar lengkap parameter dan aturan validasi, lihat CreateApplicationStatusCheck dan AssociateApplicationStatusCheck di Refer ensi API Amazon EC2.

Ambang evaluasi

`FailureThreshold`

Jumlah permintaan gagal berturut-turut sebelum cek ditandai terganggu. Standar: 2.

`SuccessThreshold`

Jumlah permintaan berhasil berturut-turut sebelum cek ditandai sehat lagi. Standar: 2.

`Timeout`

Jumlah detik untuk menunggu tanggapan sebelum permintaan dicatat sebagai gagal. Diterapkan sebagai batas waktu paksa; jika aplikasi Anda tidak merespons dalam jendela ini, permintaan dicatat sebagai kegagalan terlepas dari tanggapan akhirnya. Standar: 6. Kisaran yang valid: 1-30.

Masa tenggang awal

`InitializationGracePeriodSeconds`

Jumlah detik yang harus menunggu setelah instance diluncurkan sebelum AWS mulai mengevaluasi pemeriksaan. Gunakan parameter ini untuk memberi aplikasi waktu untuk mulai mendengarkan sebelum pemeriksaan dimulai. Jika masa tenggang terlalu singkat, Amazon EC2 Auto Scaling dapat menggantikan instans baru sebelum aplikasi mereka siap. Standar: 300. Kisaran yang valid: 1 hingga 600.

Ruang lingkup IP

`IpScope`

Pemeriksaan status aplikasi menggunakan cak `private` upan; pemeriksaan berjalan dari dalam VPC Anda. Untuk IPv4, ini sesuai dengan alamat IP pribadi instance. Untuk IPv6, AWS tidak mengklasifikasikan alamat sebagai publik atau pribadi; cek menerima alamat IPv6 apa pun dan mengevaluasinya dari dalam VPC Anda.

Indeks perangkat

`DeviceIndex`

Indeks perangkat jaringan pada instans Anda yang AWS mengevaluasi pemeriksaan kesehatan. Ubah ini jika perangkat jaringan utama instans Anda bukan perangkat yang ingin Anda periksa. Default: 0.

Agregasi, versi IP, dan jalur pemeriksaan kesehatan (subnet sumber dan tujuan serta grup keamanan) tercakup dalam bagiannya sendiri sebelumnya di halaman ini.

## Pengaturan default

Dengan AWS jalur jaringan terkelola, pemeriksaan status aplikasi menggunakan default berikut.

| Pengaturan | Default |
| --- | --- |
| Periksa interval | 60 detik (tetap; tidak dapat dikonfigurasi) |
| Ambang batas kegagalan | 2 kegagalan berturut-turut |
| Ambang batas keberhasilan | 2 keberhasilan berturut-turut |
| Batas waktu | 6 detik |
| Pencocokan kode status | 200 |
| Jalur HTTP | / |
| Versi IP | ipv4 |
| Ruang lingkup IP | private |
| Indeks perangkat | 0 |
| Masa tenggang inisialisasi | 300 detik |
| Agregasi | termasuk |
| Subnet sumber dan grup keamanan | Dikelola oleh AWS |

## Integrasi Penskalaan Otomatis Amazon EC2

Amazon EC2 Auto Scaling secara otomatis mengakhiri dan mengganti instans yang laporan status aplikasinya secara keseluruhan`impaired`, selama pemeriksaan disertakan dalam agregasi. Tidak diperlukan konfigurasi grup Penskalaan Otomatis selain mengaitkan pemeriksaan status aplikasi dengan instans dalam grup.

Amazon EC2 Auto Scaling menggunakan status keseluruhan untuk instans, bukan status pemeriksaan individual. Cek yang ditandai `excluded` tidak mendorong tindakan Penskalaan

Otomatis Amazon EC2. Pemeriksaan di `suppressed` negara bagian tidak mendorong tindakan Penskalaan Otomatis Amazon EC2.

Gunakan `InitializationGracePeriodSeconds` parameter pada pemeriksaan untuk memberikan waktu instans baru untuk memulai sebelum pemeriksaan status aplikasi dimulai. Jika masa tenggang terlalu singkat, instans baru mungkin dihentikan dan diganti dengan Amazon EC2 Auto Scaling sebelum aplikasi mereka siap melayani lalu lintas.

Pemeriksaan `InitializationGracePeriodSeconds` menetapkan berapa lama setelah instance diluncurkan sebelum pemeriksaan mulai mengevaluasi aplikasi. Atur untuk menutupi waktu startup aplikasi Anda, sehingga pemeriksaan tidak dilaporkan `impaired` saat aplikasi masih memulai. Masa tenggang pemeriksaan kesehatan grup Auto Scaling terpisah. Ini menetapkan berapa lama setelah instance memasuki layanan sebelum Amazon EC2 Auto Scaling menghentikannya karena pemeriksaan kesehatan yang gagal.

Untuk informasi selengkapnya tentang cara Amazon EC2 Auto Scaling menggunakan pemeriksaan kesehatan, lihat Pemeriksaan kesehatan untuk instans dalam grup Penskalaan Otomatis dan Menggunakan pemeriksaan status aplikasi dengan grup Penskalaan Otomatis di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

## Menangani penerapan, tambalan di tempat, dan penggantian

Penerapan, tambalan di tempat, dan operasi pemeliharaan lainnya dapat menghentikan atau memulai ulang aplikasi Anda untuk sementara waktu. Selama waktu itu, pemeriksaan status aplikasi melaporkan kegagalan karena aplikasi tidak dapat menanggapi permintaan pemeriksaan kesehatan. Jika instans Anda berada dalam grup Penskalaan Otomatis dengan pemeriksaan status aplikasi disertakan dalam agregasi, Amazon EC2 Auto Scaling mungkin menghentikan dan mengganti instans ini meskipun gangguan diharapkan terjadi.

Opsi A: Menekan cek

Gunakan penindasan untuk jendela pemeliharaan terbatas di mana Anda tahu durasinya. Penindasan diberlakukan pada tingkat instance. Anda menentukan durasi, atau menghilangkannya untuk menekan pemeriksaan sampai Anda menonaktifkan penindasan.

AWS CLI

```
aws ec2 enable-application-status-check-suppression \
        --instance-ids i-0123456789abcdef0 \
        --duration-seconds 3600
```

Respons kembali, untuk setiap contoh, ketika penekanan dimulai dan kapan akan berakhir. Keberhasilan sebagian dimungkinkan; beberapa contoh mungkin gagal ditekan dan muncul dalam respons dengan alasan.

Untuk melanjutkan pemeriksaan sebelum jendela penekanan berakhir:

```
aws ec2 disable-application-status-check-suppression \
        --instance-ids i-0123456789abcdef0
```

Saat ditekan, status aplikasi keseluruhan untuk instans dilaporkan`suppressed`. Amazon EC2 Auto Scaling tidak bertindak pada `suppressed` instans.

Opsi B: Kecualikan cek dari agregasi

Jika Anda ingin cek terus mengevaluasi dan melaporkan status individualnya tetapi tidak memengaruhi status keseluruhan atau memicu tindakan Penskalaan Otomatis Amazon EC2, setel pengaturan agregasi cek ke`excluded`. Ini berguna untuk skenario yang berumur panjang seperti meluncurkan versi pemeriksaan baru atau memvalidasi perubahan tanpa risiko penggantian, dan untuk kasus di mana Anda ingin telemetri dilanjutkan tanpa dampak operasional.

Untuk informasi selengkapnya, lihat Agregasi.

Opsi C: Putuskan hubungan cek

Gunakan disasosiasi untuk penghapusan yang berumur lebih lama atau tidak terbatas.

```
aws ec2 disassociate-application-status-check \
        --application-status-check-id asc-1234567890abcdef0 \
        --instance-ids i-0123456789abcdef0
```

Jika Anda mengaitkan berdasarkan tag, hapus tag dari instance untuk dipisahkan. Setelah disasosiasi, status aplikasi keseluruhan untuk laporan `not-applicable` instance.

Panduan penyebaran

Penerapan adalah skenario pemeliharaan paling umum yang memerlukan penindasan. Gunakan penindasan ketika alat penerapan Anda memiliki kait pra-penerapan dan kait pasca-penerapan sehingga Anda dapat menekan pemeriksaan sebelum penerapan dimulai dan menonaktifkan penindasan setelah penerapan selesai.

Pola umumnya adalah:

1. Di hook pra-penerapan, panggil enable-application-status-check-suppression untuk instance, dengan durasi yang mencakup jendela penerapan yang diharapkan.
2. Lakukan penyebaran.
3. Di hook pasca-penerapan, panggil disable-applic ation-status-check-suppression untuk instance.

Jika alat penerapan Anda tidak memiliki kait, tekan drive dari CI/CD pipeline yang memanggil penerapan.

## Menguji pemeriksaan status aplikasi baru

Anda dapat memvalidasi pemeriksaan status aplikasi baru dalam produksi sebelum mulai berkontribusi pada pemantauan tingkat instance Anda. Setel pengaturan agregasi ke `excluded` saat Anda membuat cek, lalu konfirmasikan laporan status yang diharapkan dan kode respons HTTP. Saat Anda siap, ubah setelan `included` agar pemeriksaan berkontribusi pada status keseluruhan instans dan terintegrasi dengan Amazon EC2 Auto Scaling.

1. Buat cek dengan pengaturan agregasi disetel ke`excluded`.

```
aws ec2 create-application-status-check \
        --protocol https \
        --port 443 \
        --path "/health" \
        --status-code-matcher "200" \
        --aggregation excluded
```

2. Kaitkan cek dengan instance pengujian atau subset armada produksi Anda.
3. Tunggu setidaknya dua interval pemeriksaan (sekitar dua menit) untuk memungkinkan pemeriksaan menyelesaikan evaluasi awal.
4. Gunakan describe-application-status untuk memverifikasi pemeriksaan melaporkan status yang diharapkan dan kode respons HTTP.

```
aws ec2 describe-application-status \
        --instance-ids i-0123456789abcdef0
```

5. Jika pemeriksaan dilaporkan seperti yang diharapkan, perbarui setelan agregasi `included` agar pemeriksaan berkontribusi pada status keseluruhan instans dan mendorong tindakan Penskalaan Otomatis Amazon EC2.

```
aws ec2 modify-application-status-check \
```

```
        --application-status-check-id asc-1234567890abcdef0 \
        --aggregation included
```

## Jaringan lanjutan

Pemeriksaan status aplikasi berasal dari ENI terkelola di subnet sumber dan grup keamanan yang Anda tentukan (atau yang AWS dipilih untuk Anda). Untuk beban kerja yang memerlukan ketersediaan lebih tinggi daripada yang disediakan konfigurasi sumber tunggal, atau untuk beban kerja yang berjalan di Zona Lokal atau Pos Terdepan, pertimbangkan pola berikut.

### Cross-Availability Pemantauan zona

Untuk redundansi Zona Ketersediaan, Anda dapat menjalankan pemeriksaan kesehatan dari lebih dari satu Zona Ketersediaan. Dengan jalur jaringan yang dikelola pelanggan, Anda menentukan jalur pemeriksaan kesehatan yang sumbernya berada di dua Zona Ketersediaan berbeda yang mencapai instans tujuan yang sama, menggunakan parameter. `--health-check-paths` Pemantauan dari dua Zona Ketersediaan membuat pelaporan kesehatan untuk instans Anda terus berlanjut meskipun satu Zona Ketersediaan menjadi tidak tersedia.

Contoh berikut membuat pemeriksaan dengan dua jalur pemeriksaan kesehatan yang sumbernya berada di Zona Ketersediaan yang berbeda, keduanya mencapai instans tujuan yang sama.

```
aws ec2 create-application-status-check \
        --protocol https \
        --port 443 \
        --path "/health" \
        --status-code-matcher "200" \
        --health-check-paths '[{"Source":{"SubnetId":"subnet-source-
az1","SecurityGroupId":"sg-healthcheck"},"Destinations":[{"SubnetId":"subnet-
app-az1","SecurityGroupId":"sg-app"}]},{"Source":{"SubnetId":"subnet-source-
az2","SecurityGroupId":"sg-healthcheck"},"Destinations":[{"SubnetId":"subnet-app-
az2","SecurityGroupId":"sg-app"}]}]'
```

### Zona Lokal

Untuk instans yang berjalan di Zona AWS Lokal, antarmuka jaringan elastis terkelola (ENI) berada di Wil AWS ayah induk, bukan di Zona Lokal. Lalu lintas pemeriksaan kesehatan antara Wilayah induk dan instans Zona Lokal Anda melintasi tautan layanan Zona Lokal, yang mungkin dikenakan biaya transfer data tambahan.

## Praktik terbaik

- Rancang titik akhir kesehatan Anda untuk mencerminkan kesehatan aplikasi yang berjalan pada instance tersebut. Ketika titik akhir Anda mengembalikan kesehatan berdasarkan aplikasi itu sendiri, Amazon EC2 Auto Scaling hanya menggantikan instans yang benar-benar terganggu. Jika respons titik akhir juga bergantung pada sumber daya bersama seperti database atau layanan hilir, masalah dengan sumber daya tersebut dapat gagal dalam pemeriksaan di banyak instans sekaligus. Ini dapat memicu penggantian seluruh armada. Untuk panduan penulisan titik akhir pemeriksaan kesehatan, lihat Mener apkan pemeriksaan kesehatan di Amazon Builders' Library.
- Lindungi dari kegagalan yang berkorelasi. Cek yang disertakan menggerakkan penggantian Amazon EC2 Auto Scaling. Pemeriksaan yang gagal di banyak instance sekaligus dapat memicu gelombang penggantian. Tetapkan kebijakan pemeliharaan instans pada grup Penskalaan Otomatis untuk membatasi jumlah instans yang diganti secara bersamaan. Untuk informasi selengkapnya, lihat Kebijakan pemeliharaan instans di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
- Alarm pada jumlah kasus yang terganggu. Buat CloudWatch alarm Amazon pada `StatusCheckFailed_Application` metrik di seluruh armada Anda. Peningkatan mendadak di banyak instans menunjukkan ketergantungan bersama daripada kesalahan instans individual, dan memberi Anda waktu untuk merespons sebelum penggantian jatuh. Untuk informasi selengkapnya, lihat the section called “Pantau pemeriksaan status aplikasi”.
- Tetap dalam kuota antarmuka jaringan Anda. Pemeriksaan status aplikasi membuat antarmuka jaringan terkelola yang diperhitungkan terhadap ku ota Antarmuka Jaringan per Wilayah. AWS memberlakukan kuota ini per Zona Ketersediaan. Pantau penggunaan Anda sehingga armada yang berkembang tidak mencapai kuota. Jika ya, tidak AWS dapat membuat antarmuka baru. Untuk panduan alarm kuota terkait, lihatthe section called “Kuota”.
- Perlakukan izin pemeriksaan status sebagai kontrol perubahan. Tindakan IAM yang membuat, memodifikasi, menghapus, mengaitkan, memisahkan, dan menekan pemeriksaan status aplikasi dapat memengaruhi ketersediaan instans. Tindakan ini menentukan apa yang mendorong penggantian Amazon EC2 Auto Scaling. Perlakukan tindakan seperti`ec2:CreateApplicationStatusCheck`,, `ec2:AssociateApplicationStatusCheck`ec2:`ModifyApplicationStatusCheck`, dan `ec2:EnableApplicationStatusCheckSuppression` sebagai tindakan yang dikendalikan perubahan daripada memberikannya sebagai bagian dari akses Amazon EC2 umum. Untuk daftar lengkap tindakan, lihat Referensi API Amazon EC2.

## Pemecahan masalah

Ketika pemeriksaan status aplikasi melaporkan gangguan tetapi Anda mengharapkan aplikasi Anda sehat, verifikasi masing-masing hal berikut:

1. Jangkauan contoh.  Konfirmasikan bahwa pemeriksaan status Instance dan System instans sudah adaok.
2. Aturan masuk grup keamanan.  Grup keamanan instans tujuan harus mengizinkan lalu lintas masuk pada port cek dari grup keamanan sumber yang digunakan oleh pemeriksaan status aplikasi. Untuk jalur jaringan terkel AWS ola, AWS berikan grup keamanan sumber; untuk jalur jaringan yang dikelola pelanggan, gunakan grup keamanan yang Anda tentukan sebagai sumber.
3. Firewall tuan rumah.  Setiap firewall tingkat host (iptables, Windows Firewall, firewall host pihak ketiga) pada instance harus mengizinkan lalu lintas masuk pada port cek.
4. Titik akhir aplikasi.  Aplikasi harus mendengarkan pada port dan jalur yang Anda konfigurasikan. Konfirmasikan dengan permintaan lokal dari instance (`curl http://localhost:PORT/PATH`).
5. Ketidakcocokan protokol.  Jika pemeriksaan dikonfigurasi sebagai HTTPS tetapi titik akhir hanya melayani HTTP (atau sebaliknya), semua panggilan akan gagal.
6. Pencocokan kode status.  Konfirmasikan bahwa kode respons aktual aplikasi Anda disertakan dalam pencocokan kode status yang Anda konfigurasikan.
7. Jalur jaringan.  Jika Anda mengonfigurasi jalur jaringan yang dikelola pelanggan, konfirmasikan subnet sumber dan grup keamanan memiliki konektivitas ke subnet tujuan. Gunakan VPC Reachability Analyzer untuk melacak jalur jaringan.
8. Kuota ENI yang tersedia.  AWS membuat antarmuka jaringan elastis terkelola (ENI) di akun Anda untuk setiap subnet sumber dan kombinasi grup keamanan. Konfirmasikan akun Anda belum mencapai antarmuka  Jaringan per  kuota Wilayah, yang diberlakukan per Zona Ketersediaan. Jika akun Anda telah mencapai kuota ini, AWS tidak dapat membuat ENI terkelola dan pemeriksaan tidak dapat dijalankan. Untuk informasi selengkapnya, lihat kuota Amazon VPC.

### Kode alasan

Respons deskripsi-aplikasi-status menyer takan alasan untuk setiap pemeriksaan. Alasannya berisi kode status HTTP yang dikembalikan oleh aplikasi Anda (sebagai angka), bersama dengan protokol yang digunakan untuk pemeriksaan. Cek ditandai `passed` jika kode status yang dikembalikan disertakan dalam pencocokan kode status Anda, dan `failed` sebaliknya.

Alasannya juga mencakup kode alasan dan, untuk HTTP-level hasil, protokol dan kode status HTTP yang dikembalikan. Alasannya berisi bidang berikut:

`Code`

Kode alasan untuk hasil pemeriksaan status aplikasi. Salah satu nilai berikut:

- `ResponseCodeMatched`: kode status HTTP yang dikembalikan oleh pemeriksaan kesehatan cocok dengan yang dikonfigurasi`StatusCodeMatcher`.
- `ResponseCodeMismatch`: kode status HTTP yang dikembalikan oleh pemeriksaan kesehatan tidak cocok dengan yang dikonfigurasi`StatusCodeMatcher`.
- `ConnectionTimeout`: koneksi ke target habis waktunya.
- `ResponseTimeout`: waktu pemeriksaan kesehatan habis sambil menunggu tanggapan dari target.
- `ConnectionRefused`: target menolak koneksi pemeriksaan kesehatan.
- `ConnectionReset`: koneksi pemeriksaan kesehatan diatur ulang sebelum respons diterima.

Untuk `ResponseCodeMatched` dan`ResponseCodeMismatch`, `StatusCode` bidang berisi kode status HTTP yang dikembalikan dan `Protocol` bidang berisi protokol yang digunakan untuk pemeriksaan kesehatan. Untuk kesalahan koneksi, seperti`ConnectionTimeout`, `ResponseTimeout``ConnectionRefused`,, dan`ConnectionReset`, `Protocol` bidang `StatusCode` dan tidak ada.

`Protocol`

Protokol yang digunakan untuk pemeriksaan kesehatan. Salah satu `HTTP` atau`HTTPS`.

`StatusCode`

Kode status HTTP dikembalikan oleh pemeriksaan kesehatan.

Gunakan kode status HTTP yang dikembalikan untuk mengidentifikasi mengapa pemeriksaan gagal. Beberapa contoh umum:

| Kode status HTTP | Makna khas | Remediasi umum |
|---|---|---|
| `200` | Aplikasi mengembalikan respons yang berhasil. | Tidak ada. Ini biasanya status yang sehat. |

| Kode status HTTP | Makna khas | Remediasi umum |
|---|---|---|
| `301`, `302` | Aplikasi mengembalikan pengalihan. Panggilan pemeriksaan kesehatan tidak mengikuti pengalihan. | Arahkan jalur pemeriksaan kesehatan ke tujuan pengalihan, atau tambahkan kode pengalihan ke pencocokan kode status Anda jika Anda menganggapnya sehat. |
| `401`, `403` | Aplikasi ini memerlukan otentikasi atau akses ditolak ke jalur pemeriksaan kesehatan. | Konfigurasikan jalur pemeriksaan kesehatan agar tidak diautentikasi, atau sajikan pemeriksaan kesehatan pada jalur yang tidak memerlukan kredenSIAL. |
| `404` | Jalur pemeriksaan kesehatan yang dikonfigurasi tidak ditemukan pada aplikasi. | Konfirmasikan jalur cocok dengan rute yang dilayani aplikasi Anda. |
| `500` | Aplikasi mengembalikan kesalahan server internal. | Selidiki log aplikasi pada instance. |
| `502`, `503`, `504` | Aplikasi dapat dijangkau tetapi melaporkan masalah hulu atau kapasitas. | Selidiki kesehatan aplikasi, dependensi, dan kapasitas. Jika aplikasi Anda mengembalikan kode-kode ini selama startup, tingkatkan`InitializationGracePeriodSeconds` . |

Untuk `ApplicationStatusReason` struktur lengkapnya, lihat ApplicationStatusReason di Referensi API Amazon EC2.

Kesalahan umum

- Grup keamanan tidak mengizinkan lalu lintas masuk dari sumber pemeriksaan kesehatan pada port cek.
- Aplikasi terikat `127.0.0.1` dan tidak mendengarkan pada antarmuka jaringan.

- Jalur pemeriksaan kesehatan mengembalikan pengalihan (301, 302) daripada respons sukses, dan pencocokan kode status tidak menyertakan kode pengalihan.
- Pemeriksaan dikonfigurasi untuk HTTPS tetapi aplikasi hanya melayani HTTP, atau sebaliknya.
- Aplikasi membutuhkan waktu lebih lama untuk memulai daripada `InitializationGracePeriodSeconds` nilainya, dan Amazon EC2 Auto Scaling menggantikan instance sebelum siap.

## Pantau pemeriksaan status aplikasi

Anda dapat memantau pemeriksaan status aplikasi dengan tiga cara:

- Amazon CloudWatch. Met `StatusCheckFailed_Application` rik mencerminkan status aplikasi keseluruhan untuk instance dan dapat mendorong alarm. Metrik digabungkan per instans di seluruh pemeriksaan terkait yang setelan agregasinya adalah`included`. CloudWatch juga menerbitkan metrik per cek untuk setiap cek terkait, bernama`StatusCheckFailed_Application_`*`application-status-check-id`*.
- deskripsi- instance-status. Mengembalikan status aplikasi secara keseluruhan bersama dengan informasi status instans Anda yang lain.
- deskripsi- status aplikasi. Mengembalikan hasil per instans terperinci, termasuk status individual setiap cek terkait dan kode status HTTP yang dikembalikan oleh aplikasi Anda.

Gunakan CloudWatch metrik untuk otomatisasi berbasis alarm. Gunakan `describe-instance-status` saat Anda sudah menanyainya misalnya status. Gunakan `describe-application-status` untuk visibilitas per cek terperinci.

## Keamanan dan izin

AWS membuat dan mengelola antarmuka jaringan yang digunakan untuk pemeriksaan status aplikasi melalui peran terkait layanan. Tidak diperlukan pengaturan IAM untuk layanan untuk membuat ENI ini. Peran terkait layanan menggunakan kebijakan terkel `EC2ApplicationStatusChecksServiceRolePolicy` AWS ola.

Untuk membuat, mengaitkan, mendeskripsikan, menghapus, dan menekan pemeriksaan status aplikasi sendiri, pengguna atau peran IAM Anda memerlukan izin Amazon EC2 yang sesuai. Lihat Referensi API Amazon EC2 untuk daftar tindakan lengkap.

Grup keamanan instans Anda harus mengizinkan lalu lintas masuk dari grup keamanan sumber pemeriksaan kesehatan pada port yang Anda konfigurasikan. Dengan jalur jaringan terkel AWS ola, AWS menyediakan grup keamanan sumber; dengan jalur jaringan yang dikelola pelanggan, gunakan grup keamanan yang Anda tentukan sebagai sumber.

## Harga

Pemeriksaan status aplikasi ditagih pada komponen berikut:

- Biaya per jam sebesar $0,01 untuk setiap antarmuka jaringan elastis terkelola (ENI), per Zona Ketersediaan.
- CloudWatch Harga Amazon standar berlaku untuk metrik pemeriksaan status aplikasi.

## Kuota

Pemeriksaan status aplikasi tunduk pada AWS kuota layanan. Untuk nama kuota, nilai default, dan deskripsi, lihat titik akhir dan kuota Amazon EC2 di Referensi AWS Umum.

Selain kuota AWS layanan yang memengaruhi antarmuka jaringan terkelola, pemeriksaan status aplikasi memiliki kuota layanan berikut. Anda dapat melihat penggunaan dan permintaan kenaikan dari konsol Kuota Layanan.

Dalam kuota ini, target adalah satu contoh yang dipantau oleh satu pemeriksaan kesehatan. Jika lebih dari satu pemeriksaan kesehatan memantau instans, setiap pasangan instans dan pemeriksaan kesehatan dihitung sebagai target terpisah. A sosiasi adalah aturan tag tunggal atau ID instans tunggal yang Anda kaitkan dengan pemeriksaan kesehatan. Setiap aturan atau ID instance dihitung sebagai satu asosiasi, terlepas dari berapa banyak instance yang diselesaikan.

| Kuota | Default | Dapat disesuaikan |
|---|---|---|
| Pemeriksaan kesehatan per akun | 50 | Ya, secara otomatis |
| Asosiasi per pemeriksaan kesehatan | 50 | Ya, secara otomatis |
| Asosiasi per akun | 200 | Ya, secara otomatis |
| Target per akun | 5000 | Ya, atas permintaan |

Sebagian besar kenaikan kuota disetujui secara otomatis. Peningkatan ke  Target per akun memerlukan permintaan dan persetujuan manual.

> **Important**
>
> Jika jumlah target di akun Anda melebihi ku  ota  Target per akun, target yang melebihi batas tidak dipantau dan tidak melaporkan status aplikasi. Untuk menghindari kesenjangan dalam pemantauan, pertahankan jumlah target Anda dalam kuota atau minta kenaikan.

Kami menyarankan Anda membuat CloudWatch alarm Amazon pada penggunaan kuota pemeriksaan status aplikasi sehingga Anda diberi tahu sebelum mencapai kuota. Kuota Layanan menerbitkan metrik penggunaan ke `AWS/Usage` namespace di CloudWatch, yang dapat Anda gunakan untuk membuat alarm.

## Peristiwa perubahan status untuk instans Amazon EC2

Amazon EC2 mengirimkan `EC2 Instance State-change Notification` acara ke Amazon EventBridge saat status instans berubah.

Berikut adalah data contoh untuk peristiwa ini. Dalam contoh ini, instans memasuki status `pending`.

```
{
   "id":"7bf73129-1428-4cd3-a780-95db273d1602",
   "detail-type":"EC2 Instance State-change Notification",
   "source":"aws.ec2",
   "account":"123456789012",
   "time":"2021-11-11T21:29:54Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:ec2:us-east-1:123456789012:instance/i-1234567890abcdef0"
   ],
   "detail":{
      "instance-id":"i-1234567890abcdef0",
      "state":"pending"
   }
}
```

Nilai yang mungkin untuk `state` adalah:

- `pending`

- `running`
- `stopping`
- `stopped`
- `shutting-down`
- `terminated`

Saat Anda meluncurkan atau memulai sebuah instans, instans tersebut akan memasuki status `pending`, lalu status `running`. Saat Anda menghentikan sebuah instans, instans tersebut akan memasuki status `stopping`, lalu status `stopped`. Saat Anda mengakhiri sebuah instans, instans tersebut akan memasuki status `shutting-down`, lalu status `terminated`. Untuk informasi selengkapnya, lihat Perubahan status instans Amazon EC2.

## Membuat alarm yang mengirimkan email saat instans Amazon EC2 berubah status

Untuk menerima pemberitahuan email saat instans Anda berubah status, buat topik Amazon SNS lalu buat EventBridge aturan untuk `EC2 Instance State-change Notification` acara tersebut.

Cara membuat sebuah topik SNS

1. Buka konsol Amazon SNS di https://console.aws.amazon.com/sns/v3/home.
2. Di panel navigasi, pilih Pengguna.
3. Pilih Buat topik.
4. Untuk Tipe, pilih Standar.
5. Untuk Nama, masukkan nama untuk topik Anda.
6. Pilih Buat topik.
7. Pilih Buat langganan.
8. Untuk Protokol, pilih Email.
9. Untuk Titik Akhir, ketik alamat email yang bisa Anda gunakan untuk menerima pemberitahuan.
10. Pilih Buat langganan.
11. Anda akan menerima pesan email dengan baris subjek berikut: AWS Notification - Subscription Confirmation. Ikuti petunjuk untuk mengonfirmasi langganan Anda.

Untuk membuat EventBridge aturan

1. Buka EventBridge konsol Amazon di https://console.aws.amazon.com/events/.
2. Pilih Buat aturan.
3. Untuk Nama, masukkan nama untuk topik Anda.
4. Untuk Tipe aturan, pilih Aturan dengan pola peristiwa.
5. Pilih Berikutnya.
6. Untuk Pola peristiwa, lakukan hal berikut:
   a. Untuk Sumber peristiwa, pilih Layanan AWS.
   b. Untuk Layanan AWS, pilih EC2.
   c. Untuk Jenis acara, pilih Pemberitahuan Instans State-change EC2.
   d. Secara default, kami mengirim notifikasi untuk perubahan status apa pun pada instans apa pun. Anda dapat memilih status tertentu atau instans tertentu jika menginginkannya.
7. Pilih Berikutnya.
8. Tentukan target sebagai berikut:
   a. Untuk Tipe target, pilih Layanan AWS.
   b. Untuk Pilih target, pilih topik SNS.
   c. Untuk Topik, pilih topik SNS yang Anda buat pada prosedur sebelumnya.
9. Pilih Berikutnya.
10. (Opsional) Tambahkan tanda ke aturan Anda.
11. Pilih Berikutnya.
12. Pilih Buat aturan.
13. Untuk menguji aturan Anda, lakukan perubahan status. Misalnya, mulai instans yang berhenti, hentikan instans yang sedang berjalan, atau luncurkan instans. Anda akan menerima pesan email dengan baris subjek berikut: AWS Notification Message. Badan email berisi data peristiwa.

## Acara terjadwal untuk instans Amazon EC2

Untuk memastikan keandalan dan kinerja infrastruktur, Anda AWS dapat menjadwalkan acara untuk me-reboot, menghentikan, dan menghentikan instans Anda. Peristiwa ini tidak sering terjadi.

Jika salah satu instans Anda akan terpengaruh oleh acara yang dijadwalkan, AWS memberi tahu Anda terlebih dahulu melalui email, menggunakan alamat email yang terkait dengan AWS akun Anda. Email memberikan rincian tentang acara, seperti tanggal mulai dan akhir. Bergantung pada jenis acara, Anda mungkin dapat mengambil tindakan untuk mengontrol waktu acara. AWS juga mengirimkan AWS Health acara, yang dapat Anda pantau dan kelola dengan menggunakan Amazon EventBridge. Untuk informasi selengkapnya, lihat Memantau acara AWS Health dengan Amazon EventBridge.

Acara yang dijadwalkan dikelola oleh AWS. Anda tidak dapat menjadwalkan acara untuk instans Anda. Namun, Anda dapat:

- Lihat acara terjadwal untuk instans Anda.
- Sesuaikan pemberitahuan acara terjadwal untuk menyertakan atau menghapus tag dari notifikasi email.
- Menjadwalkan ulang acara tertentu yang dijadwalkan.
- Buat jendela acara khusus untuk acara terjadwal.
- Ambil tindakan saat instance dijadwalkan untuk reboot, berhenti, atau dihentikan.

Untuk memastikan bahwa Anda menerima pemberitahuan tentang acara yang dijadwalkan, verifikasi informasi kontak Anda di halaman Akun.

> **Note**
>
> Saat sebuah instans terpengaruh oleh peristiwa terjadwal, dan peristiwa tersebut merupakan bagian dari grup Auto Scaling, Amazon EC2 Auto Scaling pada akhirnya akan menggantikan peristiwa tersebut sebagai bagian dari pemeriksaan kondisinya, tanpa perlu tindakan lebih lanjut dari Anda. Untuk informasi selengkapnya tentang pemeriksaan kesehatan yang dilakukan oleh Amazon EC2 Auto Scaling, lihat Pemeriksaan kesehatan untuk instans dalam grup Penskalaan Otomatis di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

## Tipe peristiwa terjadwal

Amazon EC2 dapat membuat jenis acara terjadwal berikut untuk instans Anda, di mana peristiwa terjadi pada waktu yang dijadwalkan:

| Tipe peristiwa | Kode acara | Tindakan acara |
| --- | --- | --- |
| Penghentian instans | `instance-stop` | Pada waktu yang dijadwalkan, instans dihentikan. Saat Anda memulainya lagi, instans dimigrasikan ke host baru. Hanya berlaku untuk instans dengan volume root Amazon EBS. |
| Pensiun instans | `instance-retirement` | Pada waktu yang dijadwalkan, instans dihentikan jika memiliki volume root Amazon EBS, atau dihentikan jika memiliki volume root penyimpanan instans. |
| Mulai ulang instans | `instance-reboot` | Pada waktu yang dijadwalkan, instans diboot ulang. Instans tetap di host, dan selama reboot, host menjalani pemeliharaan. Ini dikenal sebagai reboot di tempat. |
| Boot ulang sistem | `system-reboot` | Pada waktu yang dijadwalkan, instance di-boot ulang dan dimigrasikan ke host baru. Ini dikenal sebagai migrasi reboot. |
| Pemeliharaan sistem | `system-maintenance` | Pada waktu yang dijadwalkan, instans mungkin terpengaruh untuk sementara oleh pemeliharaan jaringan atau pemeliharaan daya. |

## Tentukan jenis acara

Anda dapat memeriksa jenis acara apa yang dijadwalkan untuk instance Anda.

### Console

Untuk menentukan jenis acara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Dalam tabel, kode acara muncul di kolom Event type.
4. Untuk memfilter tabel agar hanya menampilkan peristiwa untuk instance, di bidang pencarian pilih Jenis sumber daya: instance dari daftar filter.

### AWS CLI

Untuk menentukan jenis peristiwa untuk sebuah instance

Gunakan perintah describe-instance-status. Jika instance memiliki acara terjadwal terkait, output memberikan informasi tentang acara yang dijadwalkan.

```
aws ec2 describe-instance-status \
    --instance-ids i-1234567890abcdef0 \
    --query InstanceStatuses[].Events
```

Berikut ini adalah output contoh. Kode acara yang dijadwalkan adalah`system-reboot`.

```
[
    "Events": [
        {
            "InstanceEventId": "instance-event-0d59937288b749b32",
            "Code": "system-reboot",
            "Description": "The instance is scheduled for a reboot",
            "NotAfter": "2020-03-14T22:00:00.000Z",
            "NotBefore": "2020-03-14T20:00:00.000Z",
            "NotBeforeDeadline": "2020-04-05T11:00:00.000Z"
        }
    ]
]
```

PowerShell

Untuk menentukan jenis peristiwa untuk sebuah instance

Gunakan Get-EC2InstanceStatus cmdlet. Jika instance memiliki acara terjadwal terkait, output memberikan informasi tentang acara yang dijadwalkan.

```
(Get-EC2InstanceStatus `
    -InstanceId i-1234567890abcdef0).Events
```

Berikut ini adalah output contoh. Kode acara yang dijadwalkan adalah`system-reboot`.

```
Code              : system-reboot
Description       : The instance is scheduled for a reboot
InstanceEventId   : instance-event-0d59937288b749b32
NotAfter          : 2020-03-14T22:00:00.000Z
NotBefore         : 2020-03-14T20:00:00.000Z
NotBeforeDeadline : 2020-04-05T11:00:00.000Z
```

Daftar Isi


- Kelola instans Amazon EC2 yang dijadwalkan untuk dihentikan atau dihentikan
- Kelola instans Amazon EC2 yang dijadwalkan untuk reboot
- Kelola instans Amazon EC2 yang dijadwalkan untuk pemeliharaan
- Melihat acara terjadwal yang memengaruhi instans Amazon EC2 Anda
- Menyesuaikan pemberitahuan acara terjadwal untuk instans EC2 Anda
- Menjadwalkan ulang acara terjadwal untuk instans EC2
- Buat jendela acara khusus untuk acara terjadwal yang memengaruhi instans Amazon EC2 Anda


## Kelola instans Amazon EC2 yang dijadwalkan untuk dihentikan atau dihentikan

Saat AWS mendeteksi kegagalan host yang mendasari instans yang tidak dapat diperbaiki, instans akan dijadwalkan untuk berhenti atau dihentikan, tergantung pada jenis volume root instans.

- Jika instans memiliki volume root Amazon EBS, instans dijadwalkan untuk berhenti.
- Jika instance memiliki volume root penyimpanan instance, instance dijadwalkan untuk dihentikan.

Untuk informasi selengkapnya, lihat Pensiun instans.

> **Important**
>
> Semua data yang disimpan pada volume penyimpanan instans hilang saat instans dihentikan, dihibernasi, atau diakhiri. Ini termasuk volume penyimpanan instans yang dilampirkan ke instance yang memiliki volume root EBS. Pastikan untuk menyimpan data dari volume penyimpanan instans yang mungkin Anda perlukan nanti sebelum instans dihentikan, dihibernasi, atau diakhiri.

Tindakan yang dapat Anda ambil

Tindakan yang dapat Anda lakukan untuk instans dengan volume root EBS

Saat menerima pemberitahuan `instance-stop` acara terjadwal, Anda dapat melakukan salah satu tindakan berikut:

- Tunggu penghentian terjadwal: Anda dapat menunggu instans berhenti dalam jendela pemeliharaan terjadwalnya.
- Lakukan penghentian dan mulai secara manual: Anda dapat menghentikan dan memulai instans sendiri pada waktu yang sesuai dengan Anda, yang memigrasinya ke host baru. Ini tidak sama dengan me-reboot instance. Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.
- Otomatisasi berhenti dan mulai: Anda dapat mengotomatiskan penghentian dan mulai segera sebagai respons terhadap `instance-stop` acara yang dijadwalkan. Untuk informasi selengkapnya, lihat Menjalankan operasi pada instans EC2 secara otomatis sebagai respons terhadap peristiwa di AWS Health dalam Panduan AWS Health Pengguna.

Tindakan yang dapat Anda lakukan untuk instance dengan volume root penyimpanan instans

Ketika Anda menerima pemberitahuan `system-retirement` acara terjadwal, dan Anda ingin menyimpan data Anda, Anda dapat melakukan tindakan berikut:

1. Luncurkan instance pengganti dari AMI terbaru Anda.
2. Migrasikan semua data yang diperlukan ke instance pengganti sebelum instans dijadwalkan untuk dihentikan.
3. Hentikan instance asli, atau tunggu sampai berakhir seperti yang dijadwalkan.

Untuk informasi lebih lanjut tentang tindakan yang dapat Anda lakukan, lihatPensiun instans.

## Kelola instans Amazon EC2 yang dijadwalkan untuk reboot

Ketika AWS harus melakukan tugas seperti menginstal pembaruan atau memelihara host yang mendasarinya, itu dapat menjadwalkan reboot instance. Selama reboot yang dijadwalkan, instance tetap berada di host yang sama, atau bermigrasi ke host yang berbeda, tergantung pada acara, sebagai berikut:

- Kejadian `instance-reboot`
  - Selama reboot, instance tetap di host. Ini dikenal sebagai reboot di tempat.
  - Tuan rumah saat ini menjalani pemeliharaan.
  - Biasanya selesai dalam hitungan detik.
- Kejadian `system-reboot`
  - Selama reboot, instance dimigrasikan ke host baru. Ini dikenal sebagai migrasi reboot.
  - Biasanya selesai dalam hitungan menit.

Untuk memeriksa jenis acara apa yang dijadwalkan untuk instance Anda, lihatTentukan jenis acara.

Tindakan yang dapat Anda ambil

Saat menerima pemberitahuan jadwal `instance-reboot` atau `system-reboot` acara, Anda dapat melakukan salah satu tindakan berikut:

- Tunggu reboot terjadwal: Anda dapat menunggu reboot instance terjadi dalam jendela pemeliharaan terjadwalnya.
- Menjadwal ulang reboot: Anda dapat men jadwal ulang reboot instance ke tanggal dan waktu yang sesuai dengan Anda.
- Lakukan reboot yang diprakarsai pengguna: Anda dapat me-reboot instans secara manual pada waktu yang sesuai dengan Anda. Namun, hasilnya tergantung pada acara:
  - `instance-reboot`event — Instans Anda tetap pada perangkat keras saat ini (reboot di tempat), tidak ada pemeliharaan host yang dilakukan, dan acara tetap terbuka.
  - Kejadian `system-reboot`
    - Jika migrasi reboot diaktifkan pada instans Anda, reboot yang diprakarsai pengguna akan mencoba memigrasikan instans Anda ke perangkat keras baru. Jika berhasil, acara dihapus. Jika tidak berhasil, reboot di tempat terjadi dan acara tetap dijadwalkan.

- Jika migrasi reboot dinonaktifkan pada instans Anda, reboot yang diprakarsai pengguna membuat instance tetap pada perangkat keras yang sama (reboot di tempat), tidak ada pemeliharaan host yang dilakukan, dan acara tetap dijadwalkan. Ketika acara yang dijadwalkan akhirnya terjadi, AWS akan memindahkan instance Anda ke perangkat keras baru (migrasi reboot).

Setelah AWS me-reboot instance Anda

Berikut ini berlaku setelah me- AWS reboot instance Anda:

- Acara yang dijadwalkan dihapus.
- Deskripsi acara diperbarui.
- Untuk `instance-reboot` acara:
  - Pemeliharaan host yang mendasarinya selesai.
- Untuk `system-reboot` acara:
  - Instance pindah ke host baru.
  - Instance mempertahankan alamat IP dan nama DNS-nya.
  - Setiap data pada volume penyimpanan instans lokal dipertahankan.
- Anda dapat menggunakan instance Anda setelah di-boot sepenuhnya.

Opsi alternatif

Jika Anda tidak dapat menjadwal ulang acara reboot atau mengaktifkan migrasi reboot untuk reboot yang diprakarsai pengguna, tetapi ingin mempertahankan operasi normal selama jendela pemeliharaan terjadwal, Anda dapat melakukan hal berikut:

- Untuk contoh dengan volume root EBS
  - Hentikan dan mulai instance secara manual untuk memigrasinya ke host baru. Ini tidak sama dengan me-reboot instance secara manual, di mana instance tetap di host yang sama.
  - Secara opsional, otomatiskan penghentian instans langsung dan mulai sebagai respons terhadap acara reboot yang dijadwalkan. Untuk informasi selengkapnya, lihat Menjalankan operasi pada instans EC2 secara otomatis sebagai respons terhadap peristiwa di AWS Health dalam Panduan AWS Health Pengguna.

> **Important**
>
> Data pada volume penyimpanan instans hilang saat instance dihentikan. Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.

- Untuk contoh dengan instance menyimpan volume root
  1. Luncurkan instance pengganti dari AMI terbaru Anda.
  2. Migrasikan semua data yang diperlukan ke instance pengganti sebelum jendela pemeliharaan terjadwal.
  3. Hentikan instance asli.

Mengaktifkan atau menonaktifkan migrasi reboot

Saat instance dijadwalkan untuk suatu `system-reboot` acara, Anda dapat me-reboot sebelum acara. Hasil reboot yang diprakarsai pengguna bergantung pada pengaturan migrasi reboot instans:

- Diaktifkan — Reboot yang diprakarsai pengguna mencoba untuk memigrasikan instans Anda ke perangkat keras baru (migrasi reboot). Jika berhasil, acara dihapus. Jika tidak berhasil, reboot di tempat terjadi dan acara tetap dijadwalkan. Perhatikan bahwa meskipun diaktifkan, migrasi reboot hanya dapat terjadi jika instans Anda memenuhi persyaratan migrasi reboot.
- Dinonaktifkan — Reboot yang diprakarsai pengguna membuat instance tetap pada perangkat keras yang sama (reboot di tempat), tidak ada pemeliharaan host yang dilakukan, dan acara tetap dijadwalkan. Ketika acara yang dijadwalkan akhirnya terjadi, AWS akan memindahkan instance Anda ke perangkat keras baru (migrasi reboot).

Reboot dengan migrasi membutuhkan waktu lebih lama daripada reboot di tempat:

- In-place reboot: Sekitar 30 detik
- Reboot dengan migrasi: Beberapa menit

> **Note**
>
> Instans yang menerima pemberitahuan `system-reboot` peristiwa diaktifkan untuk migrasi reboot yang diprakarsai pengguna secara default.

### Persyaratan untuk mengaktifkan migrasi reboot

Migrasi reboot dapat diaktifkan pada instans yang memenuhi kriteria berikut:

#### Tipe instans

Tidak semua jenis instans mendukung pengaktifan migrasi reboot. Anda dapat melihat jenis instans yang mendukung pengaktifan migrasi reboot.

##### Console

Untuk melihat jenis instans yang mendukung pengaktifan migrasi reboot

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Tipe Instans.
3. Di bilah filter, masukkan dukungan Migrasi Reboot: didukung. Saat Anda memasukkan karakter dan nama filter muncul, Anda dapat memilihnya.

   Tabel Jenis Instance menampilkan semua jenis instans yang mendukung pengaktifan migrasi reboot.

##### AWS CLI

Untuk melihat jenis instans yang mendukung pengaktifan migrasi reboot

Gunakan perintah describe-instance-types dengan filter. `reboot-migration-support`

```
aws ec2 describe-instance-types \
    --filters Name=reboot-migration-support,Values=supported \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

##### PowerShell

Untuk melihat jenis instans yang mendukung pengaktifan migrasi reboot

Gunakan Get-EC2InstanceType cmdlet dengan filter. `reboot-migration-support`

```
Get-EC2InstanceType `
    -Filter @{Name="reboot-migration-support";Values="true"} | `
    Select InstanceType | Sort-Object InstanceType
```

Penghunian

- Bersama
- Instans Khusus

Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2.

Batasan

Migrasi reboot tidak didukung untuk instance dengan karakteristik berikut:

- Platform: Instans berjalan secara native pada hypervisor Xen
- Ukuran instans: `metal` instance
- Penyewaan: Tuan Rumah Khusus. Untuk Host Khusus, gunakan Pemulihan Otom atis Host Khusus sebagai gantinya.
- Penyimpanan: Instans dengan volume penyimpanan instans
- Jaringan: Instans menggunakan Adaptor Kain Elastis
- Penskalaan Otomatis: Instans yang merupakan bagian dari grup Penskalaan Otomatis

Langkah-langkah untuk mengaktifkan atau menonaktifkan migrasi reboot

Ketika sebuah instance menerima `system-reboot` peristiwa, instans diaktifkan untuk migrasi reboot secara default. Anda dapat menonaktifkan migrasi reboot sehingga selama reboot yang diprakarsai pengguna, instance tetap pada perangkat keras yang sama (reboot di tempat).

`default`Konfigurasi tidak mengaktifkan migrasi reboot untuk instance yang tidak didukung. Untuk informasi selengkapnya, lihat Persyaratan untuk mengaktifkan migrasi reboot.

Anda dapat menonaktifkan atau mengaktifkan migrasi reboot pada instans yang sedang berjalan atau dihentikan.

AWS CLI

Untuk menonaktifkan migrasi reboot

Gunakan perintah modify-instance-maintenance options dan atur parameter ke. `--reboot-migration disabled`

```
aws ec2 modify-instance-maintenance-options \
```

```
    --instance-id i-0abcdef1234567890 \

    --reboot-migration disabled
```

Untuk mengaktifkan migrasi reboot

Gunakan perintah modify-instance-maintenance options dan atur parameter ke. `--reboot-migration default`

```
aws ec2 modify-instance-maintenance-options \

    --instance-id i-0abcdef1234567890 \

    --reboot-migration default
```

PowerShell

Untuk menonaktifkan migrasi reboot

Gunakan Edit-EC2InstanceMaintenanceOption cmdlet.

```
Edit-EC2InstanceMaintenanceOption `
    -InstanceId  `
    -RebootMigration Disabled
```

Untuk mengaktifkan migrasi reboot

Gunakan Edit-EC2InstanceMaintenanceOption cmdlet.

```
Edit-EC2InstanceMaintenanceOption `
    -InstanceId i-1234567890abcdef0 `
    -RebootMigration Enabled
```

## Kelola instans Amazon EC2 yang dijadwalkan untuk pemeliharaan

Ketika AWS harus mempertahankan host yang mendasarinya untuk sebuah instance, itu menjadwalkan instance untuk pemeliharaan. Terdapat dua tipe peristiwa pemeliharaan: pemeliharaan jaringan dan pemeliharaan daya.

- Selama pemeliharaan jaringan, instans terjadwal kehilangan konektivitas jaringan dalam jangka waktu singkat. Konektivitas jaringan normal ke instans Anda akan dipulihkan setelah pemeliharaan selesai.
- Selama pemeliharaan daya, instans terjadwal akan offline dalam jangka waktu singkat, lalu di-boot ulang. Saat boot ulang dilakukan, semua pengaturan konfigurasi instans Anda dipertahankan.

Setelah instans di-boot ulang (biasanya membutuhkan waktu beberapa menit), verifikasi bahwa aplikasi Anda berfungsi seperti yang diharapkan. Pada tahap ini, instans Anda seharusnya tidak lagi memiliki peristiwa terjadwal yang terkait dengannya, atau jika masih ada, deskripsi peristiwa terjadwal dimulai dengan [Completed]. Terkadang, diperlukan waktu hingga 1 jam untuk menyegarkan deskripsi status instans. Peristiwa pemeliharaan yang sudah selesai akan ditampilkan di dasbor konsol Amazon EC2 hingga seminggu.

Tindakan yang dapat Anda ambil

Tindakan yang dapat Anda lakukan untuk instans dengan volume root EBS

Saat menerima pemberitahuan `system-maintenance` acara, Anda dapat melakukan salah satu tindakan berikut:

- Tunggu pemeliharaan terjadwal: Anda dapat menunggu pemeliharaan terjadi sesuai jadwal.
- Lakukan penghentian dan penghenti an manual: Anda dapat menghentikan dan memulai instance, yang memigrasinya ke host baru. Ini tidak sama dengan me-reboot instance. Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.
- Otomatisasi berhenti dan mulai: Anda dapat mengotomatiskan penghentian dan mulai segera sebagai respons terhadap acara pemeliharaan terjadwal. Untuk informasi selengkapnya, lihat Menjalankan operasi pada instans EC2 secara otomatis sebagai respons terhadap peristiwa di AWS Health dalam Panduan AWS Health Pengguna.

Tindakan yang dapat Anda lakukan untuk instance dengan volume root penyimpanan instans

Saat menerima pemberitahuan `system-maintenance` acara, Anda dapat melakukan salah satu tindakan berikut:

- Tunggu pemeliharaan terjadwal: Anda dapat menunggu pemeliharaan terjadi sesuai jadwal.
- Luncurkan instance pengganti: Jika Anda ingin mempertahankan operasi normal selama jendela pemeliharaan terjadwal:

1. Luncurkan instance pengganti dari AMI terbaru Anda.
2. Migrasikan semua data yang diperlukan ke instance pengganti sebelum jendela pemeliharaan terjadwal.
3. Hentikan instance asli.

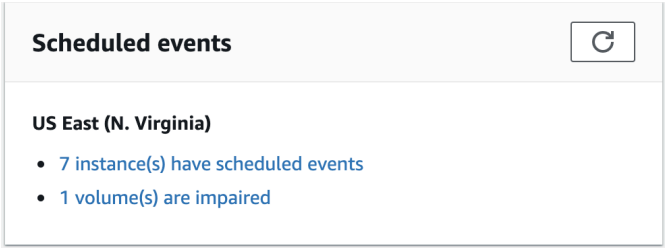
## Melihat acara terjadwal yang memengaruhi instans Amazon EC2 Anda

Selain menerima pemberitahuan acara terjadwal melalui email, Anda dapat memeriksa acara yang dijadwalkan.

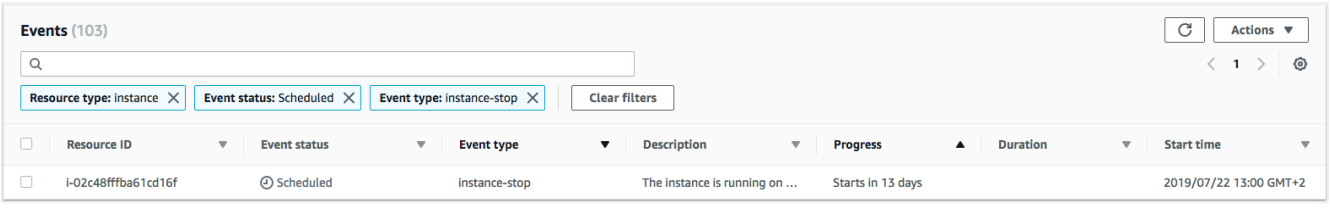
### Console

Untuk melihat peristiwa terjadwal instans Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Dasbor menampilkan semua sumber daya dengan peristiwa terkait di bagian Peristiwa terjadwal.




3. Untuk detail selengkapnya, pilih Peristiwa pada panel navigasi. Semua sumber daya dengan peristiwa terkait akan ditampilkan. Anda dapat memfilter berdasarkan karakteristik, seperti tipe peristiwa, tipe sumber daya, dan Zona Ketersediaan.




### AWS CLI

Untuk melihat peristiwa terjadwal instans Anda

Gunakan perintah describe-instance-status.

```
aws ec2 describe-instance-status \
    --instance-ids i-1234567890abcdef0 \
    --query "InstanceStatuses[].Events"
```

Contoh output berikut menunjukkan peristiwa boot ulang.

```
[
    "Events": [
        {
            "InstanceEventId": "instance-event-0d59937288b749b32",
            "Code": "system-reboot",
            "Description": "The instance is scheduled for a reboot",
            "NotAfter": "2019-03-15T22:00:00.000Z",
            "NotBefore": "2019-03-14T20:00:00.000Z",
            "NotBeforeDeadline": "2019-04-05T11:00:00.000Z"
         }

    ]
]
```

Output contoh berikut menunjukkan peristiwa retirement instans.

```
[
    "Events": [
        {
            "InstanceEventId": "instance-event-0e439355b779n26",

            "Code": "instance-stop",
            "Description": "The instance is running on degraded hardware",
            "NotBefore": "2015-05-23T00:00:00.000Z"
        }
    ]
]
```

PowerShell

Untuk melihat peristiwa terjadwal instans Anda

Gunakan perintah berikut Get-EC2InstanceStatus.

```
(Get-EC2InstanceStatus -InstanceId i-1234567890abcdef0).Events
```

Contoh output berikut menunjukkan peristiwa pemensiunan instans.

```
Code         : instance-stop
Description  : The instance is running on degraded hardware
NotBefore    : 5/23/2015 12:00:00 AM
```

Instance metadata

Untuk melihat peristiwa terjadwal pada instans Anda menggunakan metadata instans

Anda dapat mengambil informasi tentang peristiwa pemeliharaan yang aktif pada instans dari metadata instans menggunakan Layanan Metadata Instans Versi 2 atau Layanan Metadata Instans Versi 1.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/scheduled
```

Berikut adalah contoh output dengan informasi tentang peristiwa boot ulang sistem terjadwal, dalam format JSON.

```
[
  {
    "NotBefore" : "21 Jan 2019 09:00:43 GMT",
    "Code" : "system-reboot",
    "Description" : "scheduled reboot",
    "EventId" : "instance-event-0d59937288b749b32",
    "NotAfter" : "21 Jan 2019 09:17:23 GMT",
    "State" : "active"
  }
]
```

Untuk melihat riwayat terkait peristiwa yang selesai atau dibatalkan pada instans Anda menggunakan metadata instans

Anda dapat mengambil informasi tentang peristiwa yang selesai atau dibatalkan pada instans dari metadata instans menggunakan Layanan Metadata Instans Versi 2 atau Layanan Metadata Instans Versi 1.

IMDSv2

```
[ec2-user ~]$ TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600"` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/events/maintenance/history
```

IMDSv1

```
[ec2-user ~]$ curl http://169.254.169.254/latest/meta-data/events/maintenance/history
```

Berikut adalah contoh output dengan informasi tentang peristiwa boot ulang sistem yang dibatalkan, dan peristiwa boot ulang sistem yang selesai, dalam format JSON.

```
[
  {
    "NotBefore" : "21 Jan 2019 09:00:43 GMT",
    "Code" : "system-reboot",
    "Description" : "[Canceled] scheduled reboot",
    "EventId" : "instance-event-0d59937288b749b32",
    "NotAfter" : "21 Jan 2019 09:17:23 GMT",
    "State" : "canceled"
  },
  {
    "NotBefore" : "29 Jan 2019 09:00:43 GMT",
    "Code" : "system-reboot",
    "Description" : "[Completed] scheduled reboot",
    "EventId" : "instance-event-0d59937288b749b32",
    "NotAfter" : "29 Jan 2019 09:17:23 GMT",
    "State" : "completed"
  }
]
```

AWS Health

Anda dapat menggunakan Dasbor AWS Health untuk mempelajari tentang peristiwa yang dapat memengaruhi instance Anda. Meng Dasbor Health atur masalah dalam tiga kelompok: masalah

terbuka, perubahan terjadwal, dan pemberitahuan lainnya. Grup perubahan terjadwal berisi item yang sedang berlangsung atau yang akan datang.

Untuk informasi selengkapnya, lihat Mem ulai dengan Anda Dasbor AWS Health di Panduan AWS Health Pengguna.

## Menyesuaikan pemberitahuan acara terjadwal untuk instans EC2 Anda

Anda dapat menyesuaikan notifikasi peristiwa terjadwal untuk menyertakan tanda dalam notifikasi email. Hal ini memudahkan untuk mengidentifikasi sumber daya yang terpengaruh (instans atau Host Khusus) dan memprioritaskan tindakan untuk peristiwa mendatang.

Saat menyesuaikan notifikasi peristiwa untuk menyertakan tanda, Anda dapat memilih untuk menyertakan:

- Semua tanda yang terkait dengan sumber daya yang terpengaruh
- Hanya tanda tertentu yang terkait dengan sumber daya yang terpengaruh

Misalnya, Anda menetapkan tanda `application`, `costcenter`, `project`, dan `owner` ke semua instans. Anda dapat memilih untuk menyertakan semua tanda tersebut dalam notifikasi peristiwa. Atau, jika Anda hanya ingin melihat tanda `owner` dan `project` dalam notifikasi peristiwa, Anda dapat memilih untuk hanya menyertakan tanda tersebut.

Setelah Anda memilih tanda yang akan disertakan, notifikasi peristiwa akan menyertakan ID sumber daya (ID instans atau ID Host Khusus) serta kunci tanda dan pasangan nilai yang terkait dengan sumber daya yang terpengaruh.

Tugas

- Menyertakan tanda dalam notifikasi peristiwa
- Menghapus tanda dari notifikasi peristiwa
- Melihat tanda yang akan disertakan dalam notifikasi peristiwa

### Menyertakan tanda dalam notifikasi peristiwa

Tanda yang Anda pilih untuk disertakan berlaku pada semua sumber daya (instans dan Host Khusus) di Wilayah yang dipilih. Untuk menyesuaikan notifikasi peristiwa di Wilayah lain, pilih terlebih dahulu Wilayah yang diperlukan, lalu lakukan langkah-langkah berikut.

Console

Untuk menyertakan tanda dalam notifikasi peristiwa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih Tindakan, Kelola notifikasi peristiwa.
4. Aktifkan Sertakan tanda dalam notifikasi peristiwa.
5. Lakukan salah satu hal berikut, tergantung pada tanda yang ingin Anda sertakan dalam notifikasi peristiwa:
   - Untuk menyertakan semua tanda yang terkait dengan instans atau Host Khusus yang terpengaruh, pilih Sertakan semua tanda.
   - Untuk memilih tanda yang akan disertakan, klik Pilih tanda yang akan disertakan, lalu pilih atau masukkan kunci tanda.
6. Pilih Simpan.

AWS CLI

Untuk menyertakan semua tanda dalam notifikasi peristiwa

Gunakan perintah register-instance-event-notification- properties dan atur parameter ke. `IncludeAllTagsOfInstance` `true`

```
aws ec2 register-instance-event-notification-attributes \
    --instance-tag-attribute "IncludeAllTagsOfInstance=true"
```

Untuk menyertakan tanda tertentu dalam notifikasi peristiwa

Gunakan perintah register-instance-event-notification-properties dan tentukan tag yang akan disertakan dengan menggunakan parameter. `InstanceTagKeys`

```
aws ec2 register-instance-event-notification-attributes \
    --instance-tag-attribute 'InstanceTagKeys=["tag_key_1", "tag_key_2",
 "tag_key_3"]'
```

PowerShell

Untuk menyertakan semua tanda dalam notifikasi peristiwa

Gunakan Register-EC2InstanceEventNotificationAttribute cmdlet.

```
Register-EC2InstanceEventNotificationAttribute `
    -InstanceTagAttribute_IncludeAllTagsOfInstance $true
```

Untuk menyertakan tanda tertentu dalam notifikasi peristiwa

Gunakan Register-EC2InstanceEventNotificationAttribute cmdlet.

```
Register-EC2InstanceEventNotificationAttribute `
    -InstanceTagAttribute_InstanceTagKey tag_key_1, tag_key_2, tag_key_3
```

### Menghapus tanda dari notifikasi peristiwa

Anda dapat menghapus tag dari notifikasi acara.

#### Console

Untuk menghapus tanda dari notifikasi peristiwa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih Tindakan, Kelola notifikasi peristiwa.
4. Untuk menghapus semua tanda dari notifikasi peristiwa, nonaktifkan Sertakan tanda dalam notifikasi peristiwa.
5. Untuk menghapus tanda tertentu dari notifikasi peristiwa, pilih X) pada kunci tanda yang sesuai.
6. Pilih Simpan.

#### AWS CLI

Untuk menghapus semua tanda dari notifikasi peristiwa

Gunakan perintah deregister-instance-event-notification-properties dan atur parameter ke. `IncludeAllTagsOfInstance` `false`

```
aws ec2 deregister-instance-event-notification-attributes \
    --instance-tag-attribute "IncludeAllTagsOfInstance=false"
```

Untuk menghapus tag dari notifikasi acara

Gunakan perintah deregister-instance-event-notification-properties dan tentukan tag yang akan dihapus dengan menggunakan parameter. `InstanceTagKeys`

```
aws ec2 deregister-instance-event-notification-attributes \
    --instance-tag-attribute 'InstanceTagKeys=["tag_key_3"]'
```

PowerShell

Untuk menghapus semua tanda dari notifikasi peristiwa

Gunakan Unregister-EC2InstanceEventNotificationAttribute cmdlet.

```
Unregister-EC2InstanceEventNotificationAttribute `
    -InstanceTagAttribute_IncludeAllTagsOfInstance $false
```

Untuk menghapus tag dari notifikasi acara

Gunakan Unregister-EC2InstanceEventNotificationAttribute cmdlet.

```
Unregister-EC2InstanceEventNotificationAttribute `
    -InstanceTagAttribute_InstanceTagKey tag_key_3
```

Melihat tanda yang akan disertakan dalam notifikasi peristiwa

Anda dapat melihat tag yang akan disertakan dalam pemberitahuan acara.

Console

Untuk melihat tanda yang akan disertakan dalam notifikasi peristiwa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih Tindakan, Kelola notifikasi peristiwa.

AWS CLI

Untuk melihat tag yang akan disertakan dalam notifikasi acara

Gunakan perintah describe-inst ance-event-notification-properties.

```
aws ec2 describe-instance-event-notification-attributes
```

PowerShell

Untuk melihat tag yang akan disertakan dalam notifikasi acara

Gunakan Get-EC2InstanceEventNotificationAttribute cmdlet.

```
Get-EC2InstanceEventNotificationAttribute
```

## Menjadwalkan ulang acara terjadwal untuk instans EC2

Anda dapat menjadwalkan ulang peristiwa agar terjadi pada tanggal dan waktu tertentu yang sesuai untuk Anda. Setelah Anda menjadwal ulang acara, mungkin perlu satu atau dua menit sebelum tanggal yang diperbarui ditampilkan.

Batasan

- Hanya peristiwa dengan tanggal batas waktu yang dapat dijadwalkan ulang. Peristiwa dapat dijadwalkan ulang hingga tanggal batas waktu. Tanggal tenggat waktu acara ditunjukkan di kolom Batas waktu (konsol) dan `NotBeforeDeadline` bidang (AWS CLI).
- Hanya peristiwa yang belum dimulai yang dapat dijadwalkan ulang. Waktu mulai ditunjukkan di kolom Waktu mulai (konsol) dan `NotBefore` bidang (AWS CLI). Acara yang dijadwalkan dimulai dalam 5 menit ke depan tidak dapat dijadwalkan ulang.
- Waktu mulai peristiwa yang baru harus setidaknya 60 menit dari waktu saat ini.
- Jika Anda menjadwalkan ulang banyak peristiwa menggunakan konsol, tanggal batas waktu peristiwa tersebut ditentukan oleh peristiwa dengan tanggal batas waktu paling awal.

Console

Untuk menjadwalkan ulang peristiwa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih Tipe sumber daya: instans dari daftar filter.

4. Pilih satu atau beberapa instans, lalu pilih Tindakan, Jadwalkan peristiwa.

   Hanya peristiwa dengan tanggal batas waktu, yang ditunjukkan dengan nilai untuk Batas waktu, yang dapat dijadwalkan ulang. Jika salah satu peristiwa yang dipilih tidak memiliki tanggal batas waktu, Tindakan, Jadwalkan peristiwa dinonaktifkan.
5. Untuk Waktu mulai baru, masukkan tanggal dan waktu baru untuk peristiwa tersebut. Tanggal dan waktu baru harus terjadi sebelum Batas waktu peristiwa.
6. Pilih Simpan.

   Mungkin diperlukan waktu satu atau dua menit untuk menampilkan waktu mulai peristiwa yang terbaru di konsol.

AWS CLI

Untuk menjadwalkan ulang peristiwa

Gunakan perintah modify-instance-event-start-time.

```
aws ec2 modify-instance-event-start-time \
    --instance-id i-1234567890abcdef0 \
    --instance-event-id instance-event-0d59937288b749b32 \
    --not-before 2020-03-25T10:00:00.000
```

PowerShell

Untuk menjadwalkan ulang peristiwa

Gunakan Edit-EC2InstanceEventStartTime cmdlet.

```
Edit-EC2InstanceEventStartTime `
    -InstanceId i-1234567890abcdef0 `
    -InstanceEventId instance-event-0d59937288b749b32 `
    -NotBefore 2020-03-25T10:00:00.000
```

## Buat jendela acara khusus untuk acara terjadwal yang memengaruhi instans Amazon EC2 Anda

Anda dapat menentukan jendela peristiwa kustom yang berulang setiap minggu untuk peristiwa terjadwal yang melakukan boot ulang, menghentikan, atau mengakhiri instans Amazon EC2 Anda.

Anda dapat mengaitkan satu atau beberapa instans dengan jendela peristiwa. Jika peristiwa terjadwal untuk instans tersebut direncanakan, AWS akan menjadwalkan peristiwa dalam jendela peristiwa terkait.

Anda dapat menggunakan jendela peristiwa untuk memaksimalkan ketersediaan beban kerja dengan menentukan jendela peristiwa yang terjadi selama periode di luar jam sibuk untuk beban kerja Anda. Anda juga dapat menyelaraskan jendela peristiwa dengan jadwal pemeliharaan internal Anda.

Tetapkan jendela peristiwa dengan menentukan serangkaian rentang waktu. Rentang waktu minimum adalah 2 jam. Rentang waktu gabungan setidaknya harus mencapai total 4 jam.

Anda dapat mengaitkan satu atau beberapa instans dengan jendela peristiwa menggunakan ID instans atau tanda instans. Anda juga dapat mengaitkan Host Khusus dengan jendela peristiwa menggunakan ID host.

> **Warning**
>
> Jendela peristiwa hanya berlaku untuk peristiwa terjadwal yang menghentikan, melakukan boot ulang, atau mengakhiri instans.
> Jendela peristiwa tidak berlaku untuk:
>
> - Peristiwa terjadwal yang dipercepat dan peristiwa pemeliharaan jaringan.
> - Pemeliharaan tak terjadwal seperti pemulihan instans otomatis dan reboot yang tidak direncanakan.

Bekerja dengan jendela peristiwa

- Pertimbangan-pertimbangan
- Membuat jendela peristiwa
- Mengaitkan target dengan jendela peristiwa
- Memisahkan target dari jendela acara
- Memodifikasi jendela acara
- Menghapus jendela acara

Pertimbangan-pertimbangan

- Format waktu semua jendela peristiwa adalah UTC.

- Jendela acara dapat berisi beberapa rentang waktu. Sementara setiap rentang individu harus minimal 2 jam, total durasi di semua rentang harus minimal 4 jam.
- Hanya satu tipe target (ID instans, ID Host Khusus, atau tanda instans) yang dapat dikaitkan dengan suatu jendela peristiwa.
- Target (ID instans, ID Host Khusus, atau tanda instans) hanya dapat dikaitkan dengan satu jendela peristiwa.
- Maksimal 100 ID instans, atau 50 ID Host Khusus, atau 50 tanda instans dapat dikaitkan dengan suatu jendela peristiwa. Tanda instans dapat dikaitkan dengan sejumlah instans.
- Maksimal 200 jendela acara dapat dibuat per Wil AWS ayah.
- Banyak instans yang terkait dengan jendela peristiwa berpotensi memiliki peristiwa terjadwal yang terjadi pada saat bersamaan.
- Jika AWS sudah menjadwalkan acara, memodifikasi jendela acara tidak akan mengubah waktu acara yang dijadwalkan. Jika peristiwa memiliki tanggal batas waktu, Anda dapat menjadwalkan ulang peristiwa.
- Anda dapat menghentikan dan memulai instance sebelum acara yang dijadwalkan. Ini memigrasikan instance ke host baru dan menghapus acara.

### Membuat jendela peristiwa

Anda dapat membuat satu atau beberapa jendela peristiwa. Untuk setiap jendela peristiwa, Anda menentukan satu atau beberapa blok waktu. Misalnya, Anda dapat membuat jendela peristiwa dengan blok waktu yang terjadi setiap hari pada pukul 04.00 selama 2 jam. Atau, Anda dapat membuat jendela peristiwa dengan blok waktu yang terjadi pada hari Minggu mulai pukul 02.00 hingga 04.00 dan pada hari Rabu mulai pukul 03.00 hingga 05.00.

Jendela peristiwa berulang setiap pekan sampai Anda menghapusnya.

#### Console

Untuk membuat jendela peristiwa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih Tindakan, Kelola jendela peristiwa.
4. Pilih Buat jendela peristiwa instans.
5. Untuk Nama jendela peristiwa, masukkan nama deskriptif untuk jendela peristiwa tersebut.

6. Untuk Jadwal jendela peristiwa, pilih untuk menentukan blok waktu pada jendela peristiwa menggunakan pembuat jadwal cron atau dengan menentukan rentang waktu.

   - Jika Anda memilih Pembuat jadwal cron, tentukan hal berikut:
     1. Untuk Hari (UTC), tentukan hari dalam satu pekan sebagai waktu jendela peristiwa terjadi.
     2. Untuk Waktu mulai (UTC), tentukan waktu mulai jendela peristiwa.
     3. Untuk Durasi, tentukan durasi blok waktu dalam jendela peristiwa. Durasi minimum per blok waktu adalah 2 jam. Durasi minimum jendela peristiwa secara total harus sama dengan atau lebih dari 4 jam. Format waktunya adalah UTC.
   - Jika Anda memilih Rentang waktu, pilih Tambahkan rentang waktu baru, lalu tentukan hari dan waktu mulai serta hari dan waktu selesai. Ulangi untuk setiap rentang waktu. Durasi minimum per rentang waktu adalah 2 jam. Durasi minimum untuk semua rentang waktu yang digabungkan secara total harus sama dengan atau lebih dari 4 jam.
7. (Opsional) Untuk detail  Target, kaitkan satu atau lebih instance dengan jendela acara. Gunakan ID instance atau tag instans untuk mengaitkan instance. Gunakan ID host untuk mengaitkan Host Khusus. Ketika target ini dijadwalkan untuk pemeliharaan, peristiwa akan terjadi selama jendela acara ini.

   Perhatikan bahwa Anda dapat membuat jendela peristiwa tanpa mengaitkan target dengan jendela tersebut. Kemudian, Anda dapat memodifikasi jendela untuk mengaitkan satu atau beberapa target.
8. (Opsional) Untuk Tanda jendela peristiwa, pilih Tambahkan tanda, lalu masukkan kunci dan nilai untuk tanda tersebut. Ulangi hal itu untuk setiap tanda.
9. Pilih Buat jendela peristiwa.

AWS CLI

Untuk membuat jendela peristiwa dengan rentang waktu

Gunakan perintah create-inst ance-event-window.

```
aws ec2 create-instance-event-window \
    --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8 \
    --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
    --name myEventWindowName
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Untuk membuat jendela peristiwa dengan ekspresi cron

Gunakan perintah create-inst ance-event-window.

```
aws ec2 create-instance-event-window \
    --cron-expression "* 21-23 * * 2,3" \
    --tag-specifications "ResourceType=instance-event-
window,Tags=[{Key=K1,Value=V1}]" \
    --name myEventWindowName
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "State": "creating",
```

```
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

PowerShell

Untuk membuat jendela peristiwa dengan rentang waktu

Gunakan New-EC2InstanceEventWindow cmdlet.

```
$timeRange = New-Object Amazon.EC2.Model.InstanceEventWindowTimeRangeRequest
$timeRange.StartWeekDay = "monday"
$timeRange.EndWeekDay = "wednesday"
$timeRange.StartHour = 2
$timeRange.EndHour = 8
$tag = @{Key="key1"; Value="value1"}
$tagspec = New-Object Amazon.EC2.Model.TagSpecification
$tagspec.ResourceType = "instance-event-window"
$tagspec.Tags.Add($tag)
New-EC2InstanceEventWindow `
    -Name my-event-window `
    -TagSpecification $tagspec `
    -TimeRange @($timeRange)
```

Berikut ini adalah output contoh.

```
AssociationTarget     :
CronExpression        :
InstanceEventWindowId : iew-0abcdef1234567890
Name                  : my-event-window
State                 : creating
Tags                  : {key1}
TimeRanges            : {Amazon.EC2.Model.InstanceEventWindowTimeRange}
```

Untuk membuat jendela peristiwa dengan ekspresi cron

Gunakan New-EC2InstanceEventWindow cmdlet.

```
$tag = @{Key="key1"; Value="value1"}
$tagspec = New-Object Amazon.EC2.Model.TagSpecification
$tagspec.ResourceType = "instance-event-window"
$tagspec.Tags.Add($tag)
New-EC2InstanceEventWindow `
    -Name my-event-window `
    -TagSpecification $tagspec`
    -CronExpression "* 21-23 * * 2,3"
```

Berikut ini adalah output contoh.

```
AssociationTarget     :
CronExpression        : * 21-23 * * 2,3
InstanceEventWindowId : iew-0abcdef1234567890
Name                  : my-event-window
State                 : creating
Tags                  : {key1}
TimeRanges            : {}
```

### Mengaitkan target dengan jendela peristiwa

Setelah Anda membuat jendela acara, Anda dapat mengaitkan target dengan jendela acara. Anda hanya dapat mengaitkan satu jenis target dengan jendela acara. Anda dapat menentukan ID instance, ID Host Khusus, atau tag instans.

#### Console

Untuk mengaitkan target dengan jendela acara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih jendela acara untuk dimodifikasi.
4. Pilih T ind akan, Ubah jendela acara instance.

#### AWS CLI

Untuk mengaitkan tanda instans dengan jendela peristiwa

Gunakan perintah associate-inst ance-event-window.

```
aws ec2 associate-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [
                {
                    "Key": "k2",
                    "Value": "v2"
                },
                {
                    "Key": "k1",
                    "Value": "v1"
                }
            ],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

Untuk mengaitkan ID instance dengan jendela acara

Gunakan perintah associate-inst ance-event-window.

```
aws ec2 associate-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
```

```
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [
                "i-1234567890abcdef0",
                "i-0598c7d356eba48d7"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

Untuk mengaitkan Host Khusus dengan jendela peristiwa

Gunakan perintah associate-inst ance-event-window.

```
aws ec2 associate-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "DedicatedHostIds=h-029fa35a02b99801d"
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": [
                "h-029fa35a02b99801d"
            ]
        },
        "State": "creating"
    }
}
```

PowerShell

Untuk mengaitkan tanda instans dengan jendela peristiwa

Gunakan Register-EC2InstanceEventWindow cmdlet.

```
$tag1 = @{Key="key1"; Value="value1"}
$tag2 = @{Key="key2"; Value="value2"}
Register-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -AssociationTarget_InstanceTag @($tag1,$tag2)
```

Untuk mengaitkan ID instance dengan jendela acara

Gunakan Register-EC2InstanceEventWindow cmdlet.

```
Register-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -AssociationTarget_InstanceId i-1234567890abcdef0, i-0598c7d356eba48d7
```

Untuk mengaitkan Host Khusus dengan jendela peristiwa

Gunakan Register-EC2InstanceEventWindow cmdlet.

```
Register-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -AssociationTarget_DedicatedHostId h-029fa35a02b99801d
```

## Memisahkan target dari jendela acara

### Console

Untuk memisahkan target dengan jendela acara

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih jendela acara untuk dimodifikasi.
4. Pilih T ind akan, Ubah jendela acara instance.

### AWS CLI

Untuk memisahkan tanda instans dari jendela peristiwa

Gunakan perintah disassociate-instance-event-window.

```
aws ec2 disassociate-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceTags=[{Key=k2,Value=v2},{Key=k1,Value=v1}]"
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

Untuk memisahkan ID instance dari jendela acara

Gunakan perintah disassociate-instance-event-window.

```
aws ec2 disassociate-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target "InstanceIds=i-1234567890abcdef0,i-0598c7d356eba48d7"
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
```

```
        "State": "creating"
    }
}
```

Untuk memisahkan Host Khusus dari jendela peristiwa

Gunakan perintah disassociate- instance-event-window.

```
aws ec2 disassociate-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890 \
    --association-target DedicatedHostIds=h-029fa35a02b99801d
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating"
    }
}
```

PowerShell

Untuk memisahkan tanda instans dari jendela peristiwa

Gunakan Unregister-EC2InstanceEventWindow cmdlet.

```
$tag1 = @{Key="key1"; Value="value1"}
$tag2 = @{Key="key2"; Value="value2"}
Unregister-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -AssociationTarget_InstanceTag @($tag1, $tag2)
```

Untuk memisahkan ID instance dari jendela acara

Gunakan Unregister-EC2InstanceEventWindow cmdlet.

```
Unregister-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -AssociationTarget_InstanceId i-1234567890abcdef0, i-0598c7d356eba48d7
```

Untuk memisahkan Host Khusus dari jendela peristiwa

Gunakan Unregister-EC2InstanceEventWindow cmdlet.

```
Unregister-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -AssociationTarget_DedicatedHostId h-029fa35a02b99801d
```

### Memodifikasi jendela acara

Anda dapat memodifikasi semua bidang jendela peristiwa kecuali ID-nya. Misalnya, saat musim panas dimulai, Anda mungkin ingin mengubah jadwal jendela peristiwa. Untuk jendela peristiwa yang ada, Anda mungkin ingin menambahkan atau menghapus target.

Anda dapat memodifikasi rentang waktu atau ekspresi cron saat memodifikasi jendela peristiwa, tetapi tidak keduanya.

#### Console

Untuk memodifikasi jendela peristiwa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih Tindakan, Kelola jendela peristiwa.
4. Pilih jendela peristiwa yang akan dimodifikasi, lalu pilih Tindakan, Modifikasi jendela peristiwa instans.
5. Modifikasi bidang pada jendela peristiwa, lalu pilih Modifikasi jendela peristiwa.

#### AWS CLI

Untuk memodifikasi rentang waktu jendela peristiwa

Gunakan perintah modify- instance-event-window.

```
aws ec2 modify-instance-event-window
    --instance-event-window-id iew-0abcdef1234567890 \
    --time-range StartWeekDay=monday,StartHour=2,EndWeekDay=wednesday,EndHour=8
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Untuk memodifikasi serangkaian rentang waktu pada jendela peristiwa

Gunakan perintah modify- instance-event-window.

```
aws ec2 modify-instance-event-window
```

```
    --instance-event-window-id iew-0abcdef1234567890 \
    --time-range '[{"StartWeekDay": "monday", "StartHour": 2, "EndWeekDay":
 "wednesday", "EndHour": 8},
   {"StartWeekDay": "thursday", "StartHour": 2, "EndWeekDay": "friday",
 "EndHour": 8}]'
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "TimeRanges": [
            {
                "StartWeekDay": "monday",
                "StartHour": 2,
                "EndWeekDay": "wednesday",
                "EndHour": 8
            },
            {
                "StartWeekDay": "thursday",
                "StartHour": 2,
                "EndWeekDay": "friday",
                "EndHour": 8
            }
        ],
        "Name": "myEventWindowName",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

Untuk memodifikasi ekspresi cron jendela peristiwa

Gunakan perintah modify- instance-event-window.

```
aws ec2 modify-instance-event-window
    --instance-event-window-id iew-0abcdef1234567890 \
    --cron-expression "* 21-23 * * 2,3"
```

Berikut ini adalah output contoh.

```
{
    "InstanceEventWindow": {
        "InstanceEventWindowId": "iew-0abcdef1234567890",
        "Name": "myEventWindowName",
        "CronExpression": "* 21-23 * * 2,3",
        "AssociationTarget": {
            "InstanceIds": [
                "i-0abcdef1234567890",
                "i-0be35f9acb8ba01f0"
            ],
            "Tags": [],
            "DedicatedHostIds": []
        },
        "State": "creating",
        "Tags": [
            {
                "Key": "K1",
                "Value": "V1"
            }
        ]
    }
}
```

PowerShell

Untuk memodifikasi rentang waktu jendela peristiwa

Gunakan Edit-EC2InstanceEventWindow cmdlet.

```
$timeRange1 = New-Object Amazon.EC2.Model.InstanceEventWindowTimeRangeRequest
$timeRange1.StartWeekDay = "monday"
$timeRange1.EndWeekDay = "wednesday"
$timeRange1.StartHour = 2
```

```
$timeRange1.EndHour = 8
$timeRange2 = New-Object Amazon.EC2.Model.InstanceEventWindowTimeRangeRequest
$timeRange2.StartWeekDay = "thursday"
$timeRange2.EndWeekDay = "friday"
$timeRange2.StartHour = 1
$timeRange2.EndHour = 6
Edit-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -TimeRange @($timeRange1, $timeRange2)
```

Untuk memodifikasi ekspresi cron jendela peristiwa

Gunakan Edit-EC2InstanceEventWindow cmdlet.

```
Edit-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -CronExpression "* 21-23 * * 2,3"
```

### Menghapus jendela acara

Anda dapat menghapus satu jendela acara pada satu waktu.

#### Console

Untuk menghapus jendela peristiwa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Peristiwa.
3. Pilih Tindakan, Kelola jendela peristiwa.
4. Pilih jendela peristiwa yang akan dihapus, lalu pilih Tindakan, Hapus jendela peristiwa instans.
5. Saat diminta, masukkan **delete**, lalu pilih Hapus.

#### AWS CLI

Untuk menghapus jendela peristiwa

Gunakan perintah delete-instance-event-window dan tentukan jendela peristiwa yang akan dihapus.

```
aws ec2 delete-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890
```

Untuk menghapus paksa jendela acara

Gunakan parameter `--force-delete` jika jendela peristiwa saat ini dikaitkan dengan target.

```
aws ec2 delete-instance-event-window \
    --instance-event-window-id iew-0abcdef1234567890 \
    --force-delete
```

PowerShell

Untuk menghapus jendela peristiwa

Gunakan Remove-EC2InstanceEventWindow cmdlet.

```
Remove-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890
```

Untuk menghapus paksa jendela acara

Gunakan Remove-EC2InstanceEventWindow cmdlet.

```
Remove-EC2InstanceEventWindow `
    -InstanceEventWindowId iew-0abcdef1234567890 `
    -ForceDelete $true
```

# Memantau instans Anda menggunakan CloudWatch

Anda dapat memantau instans menggunakan Amazon CloudWatch, yang mengumpulkan dan memproses data mentah dari Amazon EC2 menjadi metrik yang dapat dibaca dan hampir real-time. Statistik ini dicatat untuk jangka waktu 15 bulan, sehingga Anda dapat mengakses informasi historis dan mendapatkan perspektif yang lebih baik tentang performa aplikasi atau layanan web Anda.

Secara default, Amazon EC2 mengirimkan data metrik CloudWatch dalam periode 5 menit. Untuk mengirim data metrik instans Anda CloudWatch dalam periode 1 menit, Anda dapat mengaktifkan pemantauan terperinci pada instans. Untuk informasi selengkapnya, lihat Kelola pemantauan terperinci untuk instans EC2 Anda.

Konsol Amazon EC2 menampilkan serangkaian grafik berdasarkan data mentah dari Amazon CloudWatch. Tergantung pada kebutuhan Anda, Anda mungkin lebih suka mendapatkan data untuk instans Anda dari Amazon CloudWatch daripada grafik di konsol.

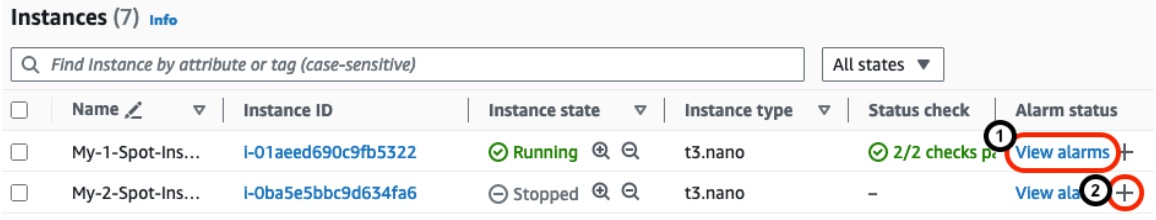
Untuk informasi CloudWatch penagihan dan biaya Amazon, lihat CloudWatch penagihan dan biaya di Panduan CloudWatch Pengguna Amazon.

Daftar Isi


- Mengelola CloudWatch alarm untuk instans EC2 Anda di konsol Amazon EC2
- Kelola pemantauan terperinci untuk instans EC2 Anda
- CloudWatch metrik yang tersedia untuk instans Anda
- Instal dan konfigurasikan CloudWatch agen menggunakan konsol Amazon EC2 untuk menambahkan metrik tambahan
- Statistik untuk CloudWatch metrik untuk instans Anda
- Melihat grafik pemantauan untuk instans Anda
- Membuat CloudWatch alarm untuk sebuah instance
- Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans


## Mengelola CloudWatch alarm untuk instans EC2 Anda di konsol Amazon EC2

Dari layar Instans di konsol Amazon EC2, Anda dapat mengelola CloudWatch alarm Amazon untuk instans Anda. Dalam tabel Instans, kolom status Alarm menyediakan dua kontrol konsol: kontrol untuk melihat alarm, dan kontrol lain untuk membuat atau mengeditnya. Tangkapan layar berikut menunjukkan kontrol konsol ini, bernomor 1 (Lihat alarm) dan 2 (tanda + untuk membuat atau mengedit alarm).




### Melihat alarm dari layar Instans

Anda dapat melihat alarm setiap instans dari layar Instans.

Untuk melihat alarm instans dari layar Instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Dalam tabel Instans, untuk instance yang Anda pilih, pilih Lihat alarm (bernomor 1 di tangkapan layar sebelumnya).
4. Di *i-1234567890abcdef0* jendela Detail alarm untuk, pilih nama alarm untuk melihat alarm di CloudWatch konsol.

## Membuat alarm dari layar Instans

Anda dapat membuat alarm untuk setiap instans dari layar Instans.

Untuk membuat alarm untuk instance dari layar Instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Dalam tabel Instans, untuk instance yang Anda pilih, pilih tanda plus (bernomor 2 di tangkapan layar sebelumnya).
4. Di layar Kelola CloudWatch alarm, buat alarm Anda. Untuk informasi selengkapnya, lihat Membuat CloudWatch alarm untuk sebuah instance.

## Mengedit alarm dari layar Instans

Anda dapat mengedit alarm untuk instance dari layar Instans.

Untuk mengedit alarm untuk instance dari layar Instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Dalam tabel Instans, untuk instance yang Anda pilih, pilih tanda plus (bernomor 2 di tangkapan layar sebelumnya).
4. Di layar Kelola CloudWatch alarm, edit alarm Anda. Untuk informasi selengkapnya, lihat Meng edit atau menghapus CloudWatch alarm di Panduan CloudWatch Pengguna Amazon.

# Kelola pemantauan terperinci untuk instans EC2 Anda

Amazon CloudWatch menyediakan dua kategori pemantauan: pemantauan dasar dan pemantauan terperinci. Secara default, instans Anda dikonfigurasi untuk pemantauan dasar. Secara opsional, Anda dapat mengaktifkan pemantauan terperinci untuk membantu Anda mengidentifikasi dan bertindak atas masalah operasional dengan lebih cepat. Anda dapat mengaktifkan atau menonaktifkan pemantauan terperinci saat peluncuran atau saat instance sedang berjalan atau dihentikan.

Mengaktifkan pemantauan terperinci pada instans tidak mempengaruhi pemantauan volume EBS yang terlampir. Untuk informasi selengkapnya, lihat CloudWatch Metrik Amazon untuk Amazon EBS.

Tabel berikut menyoroti perbedaan antara pemantauan dasar dan pemantauan terperinci untuk instans Anda.

| Tipe pemantauan | Deskripsi | Biaya |
|---|---|---|
| Pemantauan dasar | Metrik pemeriksaan status tersedia dalam periode 1 menit. Semua metrik lainnya tersedia dalam periode 5 menit. | Tidak dikenai biaya. |
| Pemantauan terperinci | Anda bisa mendapatkan metrik dalam periode 1 menit, asalkan Anda mengaktifkan pemantauan terperinci untuk instans tersebut.<br><br>Setelah mengaktifkan pemantauan terperinci, Anda dapat menggabungkan data di seluruh grup instans serupa. | Anda dikenakan biaya per metrik yang dikirim Amazon EC2. CloudWatch Anda tidak dikenai biaya untuk penyimpanan data. Untuk informasi selengkapnya, lihat Tingkat bayar di halaman CloudWatch harga Amazon. |

Daftar Isi


- Izin yang diperlukan
- Aktifkan pemantauan terperinci saat peluncuran
- Kelola pemantauan terperinci

## Izin yang diperlukan

Untuk mengaktifkan pemantauan terperinci pada instans, pengguna Anda harus memiliki izin untuk menggunakan tindakan API MonitorInstances. Untuk menonaktifkan pemantauan terperinci pada instans, pengguna Anda harus memiliki izin untuk menggunakan tindakan API UnmonitorInstances.

## Aktifkan pemantauan terperinci saat peluncuran

Gunakan prosedur berikut untuk mengaktifkan pemantauan terperinci saat peluncuran. Secara default, instans Anda menggunakan pemantauan dasar.

Console

Untuk mengaktifkan pemantauan terperinci saat meluncurkan suatu instans

Saat meluncurkan instance menggunakan konsol Amazon EC2, di bawah Detail lanjutan, pilih kotak centang Pem CloudWatch antauan terperinci.

AWS CLI

Untuk mengaktifkan pemantauan terperinci saat meluncurkan suatu instans

Gunakan perintah run-instance dengan opsi. `--monitoring`

```
--monitoring Enabled=true
```

PowerShell

Untuk mengaktifkan pemantauan terperinci saat meluncurkan suatu instans

Gunakan New-EC2Instance cmdlet dengan parameter. `-Monitoring`

```
-Monitoring $true
```

## Kelola pemantauan terperinci

Gunakan prosedur berikut untuk mengelola pemantauan terperinci untuk instans yang sedang berjalan atau dihentikan.

Console

Untuk mengelola pemantauan terperinci

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans.
4. Pilih T indakan, Monitor dan pemecahan masalah, Kel ola pemantauan terperinci.
5. Pada halaman Pemantauan terperinci, untuk Pemantauan terperinci, lakukan salah satu hal berikut:
   - Pemantauan terperinci - Pilih Aktifkan.
   - Pemantauan dasar - Hapus Aktifkan.
6. Pilih Konfirmasi.

AWS CLI

Untuk mengaktifkan pemantauan terperinci

Gunakan perintah monitor-instance berikut.

```
aws ec2 monitor-instances --instance-ids i-1234567890abcdef0
```

Untuk menonaktifkan pemantauan terperinci

Gunakan perintah unmonitor-instance.

```
aws ec2 unmonitor-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk mengaktifkan pemantauan terperinci

Gunakan Start-EC2InstanceMonitoring cmdlet.

```
Start-EC2InstanceMonitoring -InstanceId i-1234567890abcdef0
```

Untuk menonaktifkan pemantauan terperinci

Gunakan Stop-EC2InstanceMonitoring cmdlet.

```
Stop-EC2InstanceMonitoring -InstanceId i-1234567890abcdef0
```

## CloudWatch metrik yang tersedia untuk instans Anda

Amazon EC2 mengirimkan metrik ke Amazon CloudWatch. Anda dapat menggunakan Konsol Manajemen AWS, the AWS CLI, atau API untuk membuat daftar metrik yang dikirim Amazon EC2. CloudWatch Secara default, setiap titik data mencakup 5 menit yang mengikuti waktu mulai aktivitas untuk instans. Jika Anda telah mengaktifkan pemantauan terperinci, setiap poin data mencakup aktivitas menit berikutnya dari waktu mulai. Perhatikan bahwa untuk statistik Minimum, Maksimum, dan Rata-rata, perincian minimum untuk metrik yang disediakan EC2 adalah 1 menit.

Untuk informasi tentang cara melihat metrik yang tersedia menggunakan Konsol Manajemen AWS atau AWS CLI, lihat Meli hat metrik yang tersedia di Panduan CloudWatch Pengguna Amazon.

Untuk informasi cara mendapatkan statistik untuk metrik tersebut, lihat Statistik untuk CloudWatch metrik untuk instans Anda.

Daftar Isi


- Metrik instans
- Metrik akselerator
- Metrik kredit CPU
- Metrik Host Khusus
- Metrik Amazon EBS untuk instans Nitro-based
- Metrik pemeriksaan status
- Metrik pencerminan lalu lintas
- Metrik grup Auto Scaling
- Dimensi metrik Amazon EC2
- Metrik penggunaan Amazon EC2


### Metrik instans

Namespace `AWS/EC2` mencakup metrik instans berikut.

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `CPUUtilization` | Persentase waktu CPU fisik yang digunakan Amazon EC2 untuk menjalankan instans EC2, yang mencakup waktu yang digunakan untuk menjalankan kode pengguna dan kode Amazon EC2.<br><br>Pada tingkat yang sangat tinggi, `CPUUtilization` adalah jumlah `CPUUtilization` tamu dan `CPUUtilization` hypervisor.<br><br>Alat di sistem operasi Anda dapat menunjukkan persentase yang berbeda dibandingkan CloudWatch karena faktor-faktor seperti simulasi perangkat lama, konfigurasi perangkat non-warisan, beban kerja yang berat interupsi, migrasi langsung, dan pembaruan langsung. | Percent | • Average<br>• Minimum<br>• Maximum |
| `DiskReadOps` | Operasi baca yang diselesaikan dari semua volume penyimpanan instans yang tersedia untuk instans tersebut dalam periode waktu yang ditentukan.<br><br>Untuk menghitung I/O operasi rata-rata per detik (IOPS) untuk periode tersebut, bagilah total operasi dalam periode dengan jumlah detik dalam periode tersebut.<br><br>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan. | Count | • Sum<br>• Average<br>• Minimum<br>• Maximum |
| `DiskWriteOps` | Operasi tulis yang diselesaikan ke semua volume penyimpanan instans yang tersedia untuk instans tersebut dalam periode waktu yang ditentukan. | Count | • Sum<br>• Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| | Untuk menghitung I/O operasi rata-rata per detik (IOPS) untuk periode tersebut, bagilah total operasi dalam periode dengan jumlah detik dalam periode tersebut.<br><br>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan. | | |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `DiskReadBytes` | Bita yang dibaca dari semua volume penyimpanan instans yang tersedia untuk instans.<br><br>Metrik ini digunakan untuk menentukan volume data yang dibaca aplikasi dari hard disk instans. Metrik ini dapat digunakan untuk menentukan kecepatan aplikasi.<br><br>Jumlah yang dilaporkan adalah jumlah byte yang diterima selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menemukan Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `DiskReadBytes m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik bytes/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.<br><br>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan. | Bytes | • Sum<br>• Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `DiskWriteBytes` | Bita yang ditulis ke semua volume penyimpanan instans yang tersedia untuk instans.<br><br>Metrik ini digunakan untuk menentukan volume data yang ditulis aplikasi ke hard disk instans. Metrik ini dapat digunakan untuk menentukan kecepatan aplikasi.<br><br>Jumlah yang dilaporkan adalah jumlah byte yang diterima selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menemukan Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `DiskWriteBytes m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik bytes/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon.<br><br>Jika tidak ada volume penyimpanan instans, nilainya adalah 0 atau metrik tidak dilaporkan. | Bytes | • Sum<br>• Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `MetadataNoToken` | Berapa kali Instance Metadata Service (IMDS) berhasil diakses menggunakan metode yang tidak menggunakan token.<br><br>Metrik ini digunakan untuk menentukan apakah ada proses yang mengakses metadata instans yang menggunakan Instance Metadata Service Version 1 (IMDSv1), yang tidak menggunakan token. Jika semua permintaan menggunakan sesi yang didukung token, yaitu, Instance Metadata Service Version 2 (IMDSv2), nilainya adalah 0. Untuk informasi selengkapnya, lihat Transisi ke penggunaan Layanan Metadata Instans Versi 2. | Count | • Sum<br>• Percentiles |
| `MetadataNoTokenRejected` | Berapa kali panggilan IMDSv1 dicoba setelah IMDSv1 dinonaktifkan.<br><br>Jika metrik ini muncul, ini menunjukkan bahwa panggilan IMDSv1 dicoba dan ditolak. Anda dapat mengaktifkan kembali IMDSv1 atau memastikan semua panggilan Anda menggunakan IMDSv2. Untuk informasi selengkapnya, lihat Transisi ke penggunaan Layanan Metadata Instans Versi 2. | Count | • Sum<br>• Percentiles |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| NetworkIn | Jumlah bita yang diterima oleh instans di semua antarmuka jaringan. Metrik ini mengidentifikasi volume lalu lintas jaringan yang masuk ke instans tunggal.<br><br>Jumlah yang dilaporkan adalah jumlah bita yang diterima selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit) dan statistiknya adalah Sum, Anda dapat membagi angka ini dengan 300 untuk menemukannya Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit) dan statistiknya adalah Sum, bagilah dengan 60. | Bytes | • Sum<br>• Average<br>• Minimum<br>• Maximum |
| NetworkOut | Jumlah bita yang dikirimkan oleh instans di semua antarmuka jaringan. Metrik ini mengidentifikasi volume lalu lintas jaringan yang keluar dari instans tunggal.<br><br>Jumlah yang dilaporkan adalah jumlah bita yang dikirimkan selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit) dan statistiknya adalah Sum, Anda dapat membagi angka ini dengan 300 untuk menemukannya Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit) dan statistiknya adalah Sum, bagilah dengan 60. | Bytes | • Sum<br>• Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| NetworkPacketsIn | Jumlah paket yang diterima oleh instans di semua antarmuka jaringan. Metrik ini mengidentifikasi volume lalu lintas yang masuk dari segi jumlah paket pada instans tunggal.<br><br>Metrik ini hanya tersedia untuk pemantauan dasar (periode 5 menit). Untuk menghitung jumlah paket per detik (PPS) yang diterima instans Anda selama 5 menit, bagi nilai Sum statistik dengan 300. | Count | • Sum<br>• Average<br>• Minimum<br>• Maximum |
| NetworkPacketsOut | Jumlah paket yang dikirimkan oleh instans di semua antarmuka jaringan. Metrik ini mengidentifikasi volume lalu lintas yang keluar dari segi jumlah paket pada instans tunggal.<br><br>Metrik ini hanya tersedia untuk pemantauan dasar (periode 5 menit). Untuk menghitung jumlah paket per detik (PPS) yang dikirim oleh instans Anda selama 5 menit, bagilah nilai statistik Sum dengan 300. | Count | • Sum<br>• Average<br>• Minimum<br>• Maximum |

## Metrik akselerator

AWS/EC2Namespace menyertakan metrik akselerator berikut untuk inst ans komputasi yang dipercepat. Hanya didukung pada subset tipe instance komputasi yang dipercepat.

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| GPUPowerUtilization | Penggunaan daya aktif sebagai persentase daya aktif maksimum. | Persen | • Average<br>• Minimum<br>• Maximum |

## Metrik kredit CPU

Namespace `AWS/EC2` mencakup metrik kredit CPU berikut untuk instans performa yang dapat melonjak.

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
|---|---|---|---|
| `CPUCreditUsage` | Jumlah kredit CPU yang digunakan oleh instans untuk pemanfaatan CPU. Satu kredit CPU sama dengan satu vCPU yang berjalan dengan pemanfaatan 100% selama satu menit atau kombinasi yang setara dari vCPU, pemanfaatan, dan waktu (misalnya, satu vCPU yang berjalan dengan pemanfaatan 50% selama dua menit atau dua vCPU yang berjalan dengan pemanfaatan 25% selama dua menit).<br><br>Metrik kredit CPU hanya tersedia pada frekuensi 5 menit. Jika Anda menentukan periode lebih dari lima menit, gunakan statistik `Sum`, bukan statistik `Average`. | Kredit (vCPU-minutes) | • Sum<br>• Average<br>• Minimum<br>• Maximum |
| `CPUCreditBalance` | Jumlah kredit CPU yang diperoleh yang diakumulasi oleh instans sejak diluncurkan atau dimulai. Untuk T2 Standar, `CPUCreditBalance` juga mencakup jumlah kredit peluncuran yang telah diakumulasi.<br><br>Kredit diakumulasi ke saldo kredit setelah diperoleh, dan dihapus dari saldo kredit saat digunakan. Saldo kredit memiliki batas maksimum, yang ditentukan oleh ukuran instans. Setelah batas tercapai, setiap kredit yang baru diperoleh akan dibuang. Untuk T2 Standar, kredit peluncuran tidak termasuk dalam penghitungan batas. | Kredit (vCPU-minutes) | • Sum<br>• Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| | Kredit dalam `CPUCreditBalance` tersedia untuk instans untuk digunakan hingga melonjak melebihi pemanfaatan CPU acuan.<br><br>Saat sebuah instans berjalan, kredit dalam `CPUCreditBalance` tidak akan kedaluwarsa. Saat instans T3 atau T3a berhenti, nilai `CPUCreditBalance` akan bertahan selama tujuh hari. Setelah itu, semua kredit yang dikumpulkan akan hilang. Saat instans T2 berhenti, nilai `CPUCreditBalance` tidak bertahan, dan semua kredit yang masih harus dibayar hilang.<br><br>Metrik kredit CPU hanya tersedia dengan frekuensi 5 menit. | | |
| `CPUSurplusCreditBalance` | Jumlah kredit surplus yang telah digunakan oleh instans `unlimited` saat nilai `CPUCreditBalance` miliknya adalah nol.<br><br>Nilai `CPUSurplusCreditBalance` dibayarkan oleh dengan kredit CPU yang diperoleh. Jika jumlah kredit surplus melebihi jumlah kredit maksimum yang dapat diperoleh instans dalam jangka waktu 24 jam, kredit surplus yang digunakan di atas jumlah maksimum akan dikenai biaya tambahan.<br><br>Metrik kredit CPU hanya tersedia dengan frekuensi 5 menit. | Kredit (vCPU-minutes) | • Sum<br>• Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `CPUSurplusCreditsCharged` | Jumlah kredit surplus yang digunakan yang tidak ditutupi oleh kredit CPU yang diperoleh, sehingga menimbulkan biaya tambahan.<br><br>Kredit surplus yang digunakan akan dikenai biaya jika salah satu dari hal berikut terjadi:<br><br>• Kredit surplus yang digunakan melampaui jumlah kredit maksimum yang bisa didapatkan oleh instans dalam periode 24 jam. Kredit surplus yang digunakan di atas jumlah maksimum akan dikenai biaya pada akhir jam.<br>• Instans dihentikan atau diakhiri.<br>• instans dialihkan dari `unlimited` ke `standard`.<br><br>Metrik kredit CPU hanya tersedia dengan frekuensi 5 menit. | Kredit (vCPU-minutes) | • Sum<br>• Average<br>• Minimum<br>• Maximum |

## Metrik Host Khusus

Namespace `AWS/EC2` mencakup metrik berikut untuk Host Khusus T3.

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `DedicatedHostCPUUtilization` | Persentase alokasi kapasitas komputasi yang saat ini digunakan oleh instans yang berjalan di Host Khusus. | Percent | • Sum<br>• Average<br>• Minimum<br>• Maximum |

## Metrik Amazon EBS untuk instans Nitro-based

`AWS/EC2`Namespace menyertakan metrik Amazon EBS tambahan untuk volume yang dilampirkan ke Nitro-based instans yang bukan instans bare metal.

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
|---|---|---|---|
| `InstanceEBSIOPSExceededCheck` | Melaporkan apakah aplikasi mencoba mendorong IOPS yang melebihi batas EBS IOPS maksimum untuk instans dalam menit terakhir. Metrik ini dapat berupa `0` (IOPS tidak terlampaui) atau 1 (IOPS terlampaui). | Tidak ada | • Jumlah<br>• Rata-rata<br>• Minimum<br>• Maksimum |
| `InstanceEBSThroughputExceededCheck` | Melaporkan apakah aplikasi mencoba mendorong throughput yang melebihi batas throughput EBS maksimum untuk instans dalam menit terakhir. Metrik ini dapat berupa `0` (throughput tidak terlampaui) atau 1 (throughput terlampaui). | Tidak ada | • Jumlah<br>• Rata-rata<br>• Minimum<br>• Maksimum |
| `EBSReadOps` | Operasi baca yang diselesaikan dari semua volume Amazon EBS yang dilampirkan ke instans dalam periode waktu yang ditentukan.<br><br>Untuk menghitung I/O operasi baca rata-rata per detik (Baca IOPS) untuk periode tersebut, bagi total operasi dalam periode dengan jumlah detik dalam periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi jumlah ini dengan 300 untuk menghitung nilai IOPS Baca. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan operasi per detik. Misalnya, jika Anda telah membuat grafik CloudWatch | Hitungan | • Jumlah<br>• Rata-rata<br>• Minimum<br>• Maksimum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| | sebagai `EBSReadOps m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik operations/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon. | | |
| `EBSWriteOps` | Operasi tulis yang diselesaikan ke semua volume EBS yang dilampirkan pada instans tersebut dalam periode waktu yang ditentukan.<br><br>Untuk menghitung I/O operasi tulis rata-rata per detik (Tulis IOPS) untuk periode tersebut, bagi total operasi dalam periode dengan jumlah detik dalam periode tersebut. Jika menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menghitung nilai IOPS Tulis. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan operasi per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `EBSWriteOps m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik operations/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon. | Hitungan | • Jumlah<br>• Rata-rata<br>• Minimum<br>• Maksimum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `EBSReadBytes` | Bita yang dibaca dari semua volume EBS yang dilampirkan ke instans dalam jangka waktu tertentu.<br><br>Jumlah yang dilaporkan adalah jumlah bita baca selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menemukan Baca Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `EBSReadBytes m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik bytes/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon. | Byte | • Jumlah<br>• Rata-rata<br>• Minimum<br>• Maksimum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `EBSWriteBytes` | Bita yang ditulis ke semua volume EBS yang dilampirkan ke instans dalam jangka waktu tertentu.<br><br>Jumlah yang dilaporkan adalah jumlah bita tulis selama periode tersebut. Jika Anda menggunakan pemantauan dasar (5 menit), Anda dapat membagi angka ini dengan 300 untuk menemukan Tulis Bytes/second. Jika Anda memiliki pemantauan terperinci (1 menit), bagilah dengan 60. Anda juga dapat menggunakan fungsi matematika CloudWatch metrik `DIFF_TIME` untuk menemukan byte per detik. Misalnya, jika Anda telah membuat grafik CloudWatch sebagai `EBSWriteBytes m1`, rumus matematika metrik `m1/(DIFF_TIME(m1))` mengembalikan metrik bytes/second tersebut. Untuk informasi selengkapnya tentang `DIFF_TIME` dan fungsi matematika metrik lainnya, lihat Menggunakan matematika metrik di Panduan CloudWatch Pengguna Amazon. | Byte | • Jumlah<br>• Rata-rata<br>• Minimum<br>• Maksimum |
| `EBSIOBalance%` | Memberikan informasi tentang persentase I/O kredit yang tersisa di bucket burst. Metrik ini hanya tersedia untuk pemantauan dasar.<br><br>Metrik ini hanya tersedia untuk beberapa ukuran instans `*.4xlarge` dan yang lebih kecil, yang melonjak hingga performa maksimum hanya selama 30 menit setidaknya setiap 24 jam sekali.<br><br>Statistik `Sum` tidak berlaku untuk metrik ini. | Persen | • Minimum<br>• Maksimum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `EBSByteBalance%` | Memberikan informasi tentang persentase kredit throughput yang tersisa di bucket lonjakan. Metrik ini hanya tersedia untuk pemantauan dasar.<br><br>Metrik ini hanya tersedia untuk beberapa ukuran instans `*.4xlarge` dan yang lebih kecil, yang melonjak hingga performa maksimum hanya selama 30 menit setidaknya setiap 24 jam sekali.<br><br>Statistik `Sum` tidak berlaku untuk metrik ini. | Persen | • Minimum<br>• Maksimum |

Untuk informasi tentang metrik yang disediakan untuk volume EBS Anda, lihat Metrik untuk volume Amazon EBS di Panduan Pengguna Amazon EBS. Untuk informasi tentang metrik yang disediakan untuk Armada EC2 dan Armada Spot Anda, lihat. Pantau Armada EC2 atau Spot Fleet Anda menggunakan CloudWatch

## Metrik pemeriksaan status

Secara default, metrik pemeriksaan status tersedia dalam frekuensi 1 menit tanpa dikenai biaya. Untuk instans yang baru diluncurkan, data metrik pemeriksaan status hanya tersedia setelah instans tersebut menyelesaikan status inisialisasi (dalam waktu beberapa menit setelah instans memasuki status `running`). Untuk informasi selengkapnya tentang pemeriksaan status EC2, lihat Pemeriksaan status untuk instans Amazon EC2.

Namespace `AWS/EC2` mencakup metrik pemeriksaan status berikut.

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| `StatusCheckFailed` | Melaporkan apakah instance telah melewati semua pemeriksaan status di menit terakhir.<br><br>Metrik ini dapat berupa `0` (lulus) atau `1` (gagal). | Count | • Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
|---|---|---|---|
| | Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya. | | |
| `StatusCheckFailed_Instance` | Melaporkan apakah instans telah melalui pemeriksaan status instan pada menit terakhir.<br>Metrik ini dapat berupa `0` (lulus) atau `1` (gagal).<br>Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya. | Count | • Average<br>• Minimum<br>• Maximum |
| `StatusCheckFailed_System` | Melaporkan apakah instans telah melalui pemeriksaan status sistem pada menit terakhir.<br>Metrik ini dapat berupa `0` (lulus) atau `1` (gagal).<br>Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya. | Count | • Average<br>• Minimum<br>• Maximum |
| `StatusCheckFailed_AttachedEBS` | Melaporkan apakah instans telah melalui pemeriksaan status EBS yang terlampir pada menit terakhir.<br>Metrik ini dapat berupa `0` (lulus) atau `1` (gagal).<br>Secara default, metrik ini tersedia dalam frekuensi 1 menit tanpa dikenai biaya. | Count | • Average<br>• Minimum<br>• Maximum |

`AWS/EBS`Namespace menyertakan metrik pemeriksaan status berikut.

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
|---|---|---|---|
| `VolumeStalledIOCheck` | Catatan: Khusus instans Nitro. Tidak dipublikasikan untuk volume yang dilampirkan ke Amazon ECS dan AWS Fargate tugas. | Tidak ada | • Average<br>• Minimum<br>• Maximum |

| Metrik | Deskripsi | Unit | Statistik yang bermakna |
| --- | --- | --- | --- |
| | Melaporkan apakah volume telah lulus atau gagal pemeriksaan IO yang macet di menit terakhir. Metrik ini dapat berupa `0` (lulus) atau `1` (gagal). | | |

## Metrik pencerminan lalu lintas

Namespace `AWS/EC2` mencakup metrik untuk lalu lintas yang dicerminkan. Untuk informasi selengkapnya, lihat Mem antau lalu lintas cermin menggunakan Amazon CloudWatch di Panduan Pencerminan Lalu L intas Amazon VPC.

## Metrik grup Auto Scaling

Namespace `AWS/AutoScaling` mencakup metrik untuk grup Auto Scaling. Untuk informasi selengkapnya, lihat Mem antau CloudWatch metrik untuk grup dan instans Penskalaan Otomatis Anda di Panduan Pengguna Penskalaan Otomatis Amazon EC2.

## Dimensi metrik Amazon EC2

Anda dapat menggunakan dimensi berikut untuk mempersempit metrik yang terdaftar pada tabel sebelumnya.

| Dimensi | Deskripsi |
| --- | --- |
| `AutoScalingGroupName` | Dimensi ini memfilter data yang Anda minta untuk semua instans dalam grup kapasitas yang ditentukan. Grup Auto Scaling adalah kumpulan instans yang Anda tentukan jika menggunakan Penskalaan Otomatis. Dimensi ini hanya tersedia untuk metrik Amazon EC2 ketika instans berada dalam grup Auto Scaling. Tersedia untuk instans dengan Pemantauan Terperinci atau Dasar yang diaktifkan. |
| `ImageId` | Dimensi ini memfilter data yang Anda minta untuk semua instans yang menjalankan Amazon Machine Image (AMI) |

| Dimensi | Deskripsi |
| --- | --- |
| | Amazon EC2 ini. Tersedia untuk instans dengan Pemantauan Terperinci yang diaktifkan. |
| `InstanceId` | Dimensi ini hanya memfilter data yang Anda minta untuk instans yang teridentifikasi. Hal ini membantu Anda menemukan instans yang tepat untuk memantau data. |
| `InstanceType` | Dimensi ini memfilter data yang Anda minta untuk semua instans yang berjalan dengan tipe instans yang ditentukan ini. Hal ini membantu Anda mengategorikan data berdasarkan tipe instans yang berjalan. Misalnya, Anda dapat membandin gkan data dari instans m1.small dan instans m1.large untuk menentukan instans yang memiliki nilai bisnis yang lebih baik bagi aplikasi Anda. Tersedia untuk instans dengan Pemantauan Terperinci yang diaktifkan. |

## Metrik penggunaan Amazon EC2

Anda dapat menggunakan metrik CloudWatch penggunaan untuk memberikan visibilitas ke dalam penggunaan sumber daya akun Anda. Gunakan metrik ini untuk memvisualisasikan penggunaan layanan Anda saat ini pada CloudWatch grafik dan dasbor.

Metrik penggunaan Amazon EC2 sesuai dengan kuota AWS layanan. Anda dapat mengonfigurasi alarm yang memberi tahu Anda saat penggunaan mendekati kuota layanan. Untuk informasi selengkapnya tentang CloudWatch integrasi dengan kuota layanan, lihat metrik AWS penggunaan di Panduan CloudWatch Pengguna Amazon.

Amazon EC2 menerbitkan metrik berikut di namespace `AWS/Usage`.

| Metrik | Deskripsi |
| --- | --- |
| `ResourceCount` | Jumlah sumber daya yang ditentukan yang berjalan di akun Anda. Sumber daya tersebut ditentukan oleh dimensi yang dikaitkan dengan metrik. |

| Metrik | Deskripsi |
| --- | --- |
| | Statistik yang paling berguna untuk metrik ini adalah `MAXIMUM`, yang merepresentasikan jumlah maksimum sumber daya yang digunakan selama periode 1 menit. |

Dimensi berikut digunakan untuk menyempurnakan metrik penggunaan yang diterbitkan oleh Amazon EC2.

| Dimensi | Deskripsi |
| --- | --- |
| `Service` | Nama AWS layanan yang berisi sumber daya. Untuk metrik penggunaan Amazon EC2, nilai untuk dimensi ini adalah `EC2`. |
| `Type` | Tipe entitas yang dilaporkan. Saat ini, satu-satunya nilai yang valid untuk metrik penggunaan Amazon EC2 adalah `Resource`. |
| `Resource` | Tipe sumber daya yang sedang berjalan. Saat ini, satu-satunya nilai yang valid untuk metrik penggunaan Amazon EC2 adalah `vCPU`, yang mengembalikan informasi tentang instans yang sedang berjalan. |
| `Class` | Kelas sumber daya yang dilacak. Untuk metrik penggunaan Amazon EC2 dengan `vCPU` sebagai nilai dimensi `Resource`, nilai yang valid adalah `Standard/OnDemand` , `F/OnDemand` , `G/OnDemand` , `Inf/OnDemand` , `P/OnDemand` , dan `X/OnDemand` .<br><br>Nilai untuk dimensi ini menentukan huruf pertama dari tipe instans yang dilaporkan oleh metrik. Misalnya, `Standard/OnDemand` mengembalikan informasi terkait semua instans yang berjalan dengan tipe yang dimulai dengan A, C, D, H, I, M, R, T, dan Z, lalu `G/OnDemand` mengembalikan informasi terkait semua instans yang berjalan dengan tipe yang dimulai dengan G. |

# Instal dan konfigurasikan CloudWatch agen menggunakan konsol Amazon EC2 untuk menambahkan metrik tambahan

| Deskripsi |
| --- |
| Menginstal dan mengonfigurasi CloudWatch agen menggunakan konsol Amazon EC2 dalam versi beta untuk Amazon EC2 dan dapat berubah. |

Secara default, Amazon CloudWatch menyediakan metrik dasar, seperti `CPUUtilization` dan`NetworkIn`, untuk memantau instans Amazon EC2 Anda. Untuk mengumpulkan metrik tambahan, Anda dapat menginstal CloudWatch agen pada instans EC2, lalu mengonfigurasi agen untuk memancarkan metrik yang dipilih. Alih-alih menginstal dan mengonfigurasi CloudWatch agen secara manual pada setiap instans EC2, Anda dapat menggunakan konsol Amazon EC2 untuk melakukan ini untuk Anda.

Anda dapat menggunakan konsol Amazon EC2 untuk menginstal CloudWatch agen pada instans Anda dan mengonfigurasi agen untuk memancarkan metrik yang dipilih.

Atau, untuk menyelesaikan proses ini secara manual, lihat Meng instal CloudWatch agen di Panduan CloudWatch Pengguna Amazon. Untuk informasi selengkapnya tentang CloudWatch agen, lihat Meng umpulkan metrik, log, dan pelacakan menggunakan CloudWatch agen.

Daftar Isi


- Prasyarat
- Cara kerjanya
- Biaya
- Instal dan konfigurasikan CloudWatch agen


## Prasyarat

Untuk menggunakan Amazon EC2 untuk menginstal dan mengkonfigurasi CloudWatch agen, Anda harus memenuhi prasyarat pengguna dan instans yang dijelaskan di bagian ini.

> **Tip**
>
> Fitur ini tidak tersedia di semua Wilayah AWS. Jika item menu yang dijelaskan dalam prosedur penginstalan di halaman ini tidak ada di konsol Amazon EC2 dan Anda fleksibel tentang di mana instans Anda berjalan, coba Wilayah lain. Jika tidak, Anda dapat menggunakan petunjuk manual di Panduan CloudWatch Pengguna Amazon untuk menginstal dan mengkonfigurasi agen.

Prasyarat pengguna

Untuk menggunakan fitur ini, pengguna atau peran konsol IAM Anda harus memiliki izin yang diperlukan untuk menggunakan Amazon EC2 dan izin IAM berikut:

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameter",
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:*:*:parameter/EC2-Custom-Metrics-*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ssm:SendCommand",
                "ssm:ListCommandInvocations",
                "ssm:DescribeInstanceInformation"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
```

```
                "iam:GetInstanceProfile",
                "iam:SimulatePrincipalPolicy"
            ],
            "Resource": "*"
        }
    ]
}
```

Prasyarat instans

- Status instans: `running`
- Sistem operasi yang didukung: Linux
- AWS Systems Manager Agen (Agen SSM): Diinstal. Dua catatan tentang Agen SSM:
  - Agen SSM sudah diinstal sebelumnya pada beberapa Amazon Machine Images (AMI) yang disediakan oleh AWS dan pihak ketiga yang tepercaya. Untuk informasi tentang AMI yang didukung dan petunjuk untuk menginstal Agen SSM, lihat Amazon Machine Images (AMI) dengan Agen SSM yang sudah diinstal sebelumnya di Panduan Pengguna. AWS Systems Manager
  - Jika Anda mengalami masalah dengan Agen SSM, lihat Memecahkan Masalah Agen SSM di Panduan AWS Systems Manager Pengguna.
- Izin IAM untuk instance: Kebijakan ter AWS kelola berikut harus ditambahkan ke peran IAM yang dilampirkan ke instance:
  - AmazonSSMManagedInstanceCore- Mengaktifkan instance untuk menggunakan Manajer Sistem untuk menginstal dan mengkonfigurasi CloudWatch agen.
  - CloudWatchAgentServerPolicy- Mengaktifkan instance untuk menggunakan CloudWatch agen untuk menulis data ke CloudWatch.

  Untuk informasi tentang cara menambahkan izin IAM ke instans Anda, lihat Menggunakan profil instans di Panduan Pengguna  IAM.

## Cara kerjanya

Sebelum Anda dapat menggunakan konsol Amazon EC2 untuk menginstal dan mengonfigurasi CloudWatch agen, Anda harus memastikan bahwa pengguna atau peran IAM Anda, dan instans tempat Anda ingin menambahkan metrik, memenuhi prasyarat tertentu. Kemudian, Anda dapat

menggunakan konsol Amazon EC2 untuk menginstal dan mengonfigurasi CloudWatch agen pada instans yang Anda pilih.

Pertama bertemu prasyarat

- Anda memerlukan izin IAM yang diperlukan — Sebelum memulai, pastikan pengguna atau peran konsol Anda memiliki izin IAM yang diperlukan untuk menggunakan fitur ini.
- Inst ans — Untuk menggunakan fitur ini, instans EC2 Anda harus instans Linux, memiliki Agen SSM diinstal, memiliki izin IAM yang diperlukan, dan berjalan.

Maka Anda bisa gunakan fitur

1. Pilih instans Anda — Di konsol Amazon EC2, Anda memilih instans untuk menginstal dan mengonfigurasi CloudWatch agen. Anda kemudian memulai proses dengan memilih Con figure CloudWatch agent.
2. Validasi Agen SSM — Amazon EC2 memeriksa apakah Agen SSM diinstal dan dimulai pada setiap instans. Setiap contoh yang gagal dalam pemeriksaan ini dikecualikan dari proses. Agen SSM digunakan untuk melakukan tindakan pada instance selama proses ini.
3. Validasi izin IAM — Amazon EC2 memeriksa apakah setiap instans memiliki izin IAM yang diperlukan untuk proses ini. Setiap contoh yang gagal dalam pemeriksaan ini dikecualikan dari proses. Izin IAM memungkinkan CloudWatch agen mengumpulkan metrik dari instance dan mengintegrasikan AWS Systems Manager untuk menggunakan Agen SSM.
4. Val CloudWatch idasi agen — Amazon EC2 memeriksa apakah CloudWatch agen diinstal dan berjalan di setiap instans. Jika ada instans yang gagal dalam pemeriksaan ini, Amazon EC2 menawarkan untuk menginstal dan memulai CloudWatch agen untuk Anda. CloudWatch Agen akan mengumpulkan metrik yang dipilih pada setiap instance setelah proses ini selesai.
5. Pilih konfigurasi metrik — Anda memilih metrik yang akan dipancarkan CloudWatch agen dari instans Anda. Setelah dipilih, Amazon EC2 menyimpan file konfigurasi di Parameter Store, di mana file tersebut tetap ada sampai proses selesai. Amazon EC2 akan menghapus file konfigurasi dari Parameter Store kecuali prosesnya terganggu. Perhatikan bahwa jika Anda tidak memilih metrik, tetapi sebelumnya Anda menambahkannya ke instans Anda, metrik tersebut akan dihapus dari instans Anda ketika proses ini selesai.
6. Per CloudWatch barui konfigurasi agen — Amazon EC2 mengirimkan konfigurasi metrik ke CloudWatch agen. Ini adalah langkah terakhir dalam proses. Jika berhasil, instans Anda dapat memancarkan data untuk metrik yang dipilih dan Amazon EC2 menghapus file konfigurasi dari Penyimpanan Parameter.

### Biaya

Metrik tambahan yang Anda tambahkan selama proses ini ditagih sebagai metrik khusus. Untuk informasi selengkapnya tentang harga CloudWatch metrik, lihat CloudWatch Harga Amazon.

### Instal dan konfigurasikan CloudWatch agen

Anda dapat menggunakan konsol Amazon EC2 untuk menginstal dan mengonfigurasi CloudWatch agen untuk menambahkan metrik tambahan.

> **Note**
>
> Setiap kali Anda melakukan prosedur ini, Anda mengganti konfigurasi CloudWatch agen yang ada. Jika Anda tidak memilih metrik yang dipilih sebelumnya, metrik tersebut akan dihapus dari instance.

Untuk menginstal dan mengkonfigurasi CloudWatch agen menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance untuk menginstal dan mengkonfigurasi CloudWatch agen.
4. Pilih T indakan, Pantau dan pemecahan masalah, Konfigur CloudWatch asikan agen.
5. Untuk setiap langkah dalam proses, baca teks konsol, lalu pilih Ber ikutnya.
6. Untuk menyelesaikan proses, pada langkah terakhir, pilih Sel esai.

Untuk memperbarui konfigurasi agen yang dibuat oleh konsol Amazon EC2

Anda dapat menyesuaikan konfigurasi yang dibuat konsol EC2 secara manual. Untuk informasi selengkapnya, lihat Membuat atau mengedit file konfigurasi CloudWatch agen secara manual di Panduan CloudWatch Pengguna Amazon.

## Statistik untuk CloudWatch metrik untuk instans Anda

Anda bisa mendapatkan statistik untuk CloudWatch metrik untuk instans Anda. Statistik adalah agregasi data metrik selama periode waktu tertentu. CloudWatch menyediakan statistik berdasarkan

titik data metrik yang disediakan oleh data kustom Anda atau disediakan oleh layanan lain di dalam AWS CloudWatch. Agregasi dilakukan menggunakan namespace, nama metrik, dimensi, dan unit titik data dari ukuran, dalam periode waktu yang Anda tentukan. Tabel berikut menjelaskan statistik yang tersedia.

| Statistik | Deskripsi |
| --- | --- |
| `Minimum` | Nilai terendah yang diamati selama periode yang ditentukan. Anda dapat menggunakan nilai ini untuk menentukan volume aktivitas yang rendah pada aplikasi Anda. |
| `Maximum` | Nilai tertinggi yang diamati selama periode yang ditentukan. Anda dapat menggunakan nilai ini untuk menentukan volume aktivitas yang tinggi pada aplikasi Anda. |
| `Sum` | Semua nilai yang dikirimkan untuk metrik yang cocok disatukan. Statistik ini dapat berguna untuk menentukan total volume metrik. |
| `Average` | Nilai dari `Sum` / `SampleCount` selama periode yang ditentukan. Dengan membandingkan statistik ini dengan `Minimum` dan `Maximum`, Anda dapat menentukan cakupan suatu metrik dan seberapa dekat rata-rata penggunaan dengan `Minimum` dan `Maximum`. Perbandingan ini membantu Anda untuk mengetahui kapan harus menambah atau mengurangi sumber daya Anda sesuai kebutuhan. |
| `SampleCount` | Hitungan (jumlah) titik data yang digunakan untuk penghitungan statistik. |
| `pNN.NN` | Nilai persentil yang ditentukan. Anda dapat menentukan persentil apa pun, menggunakan hingga dua tempat desimal (misalnya, p95.45). |

Daftar Isi


- Mendapatkan statistik untuk instans tertentu
- Mengagregasi statistik di seluruh instans
- Mengagregasi statistik menurut grup Auto Scaling
- Mengagregasi statistik menurut AMI

## Mendapatkan statistik untuk instans tertentu

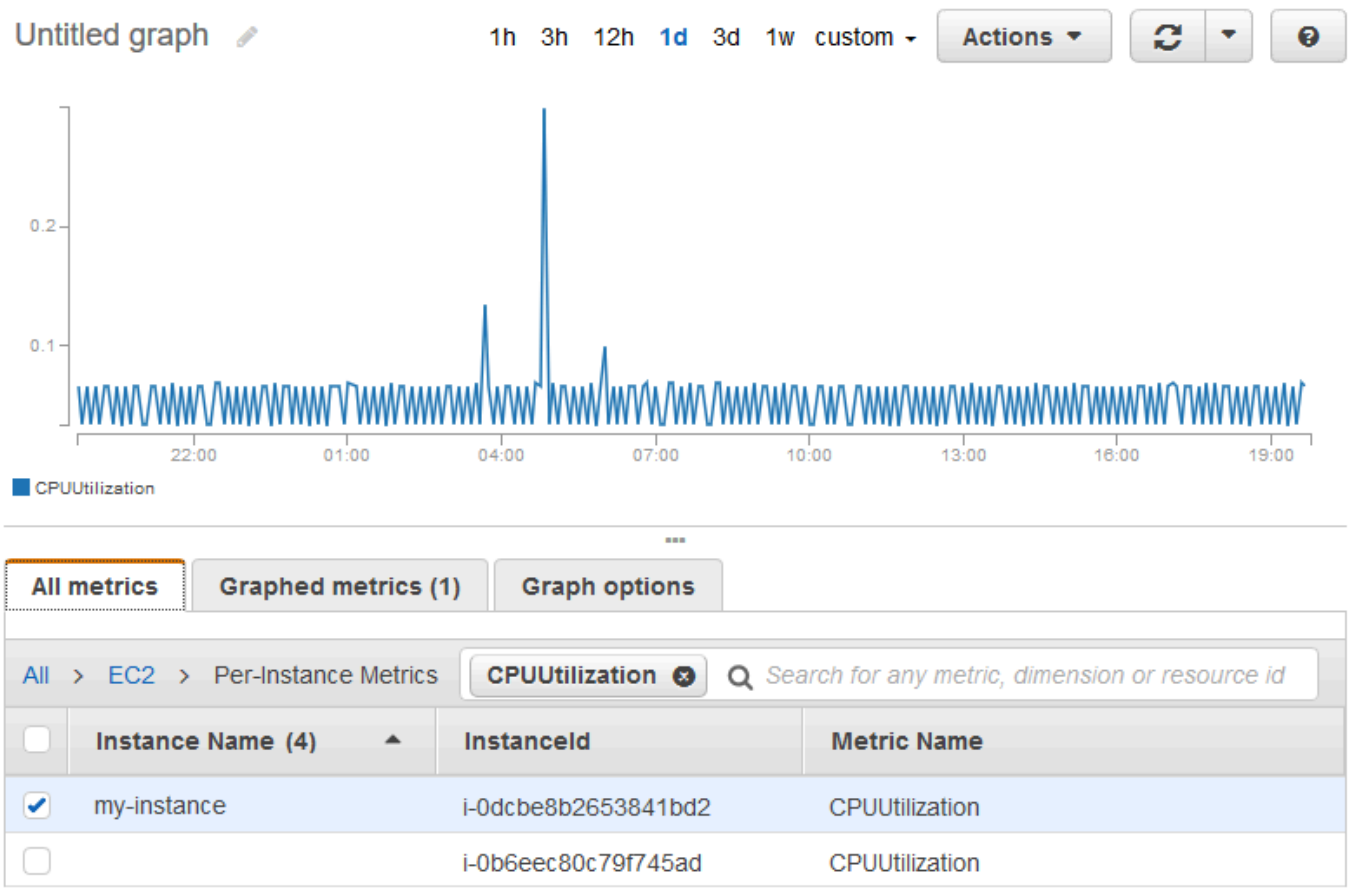
Anda dapat menggunakan Konsol Manajemen AWS atau AWS CLI untuk mendapatkan statistik untuk contoh tertentu. Contoh berikut menunjukkan cara menggunakan Konsol Manajemen AWS atau AWS CLI untuk menentukan pemanfaatan CPU maksimum instans EC2 tertentu.

Persyaratan

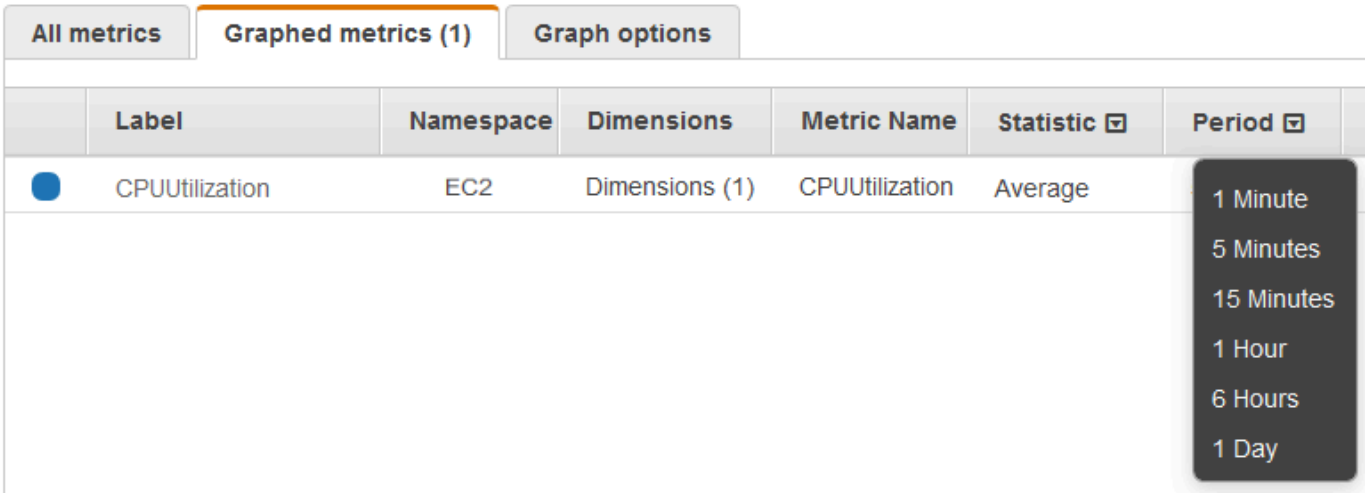
- Anda harus memiliki ID instans. Anda bisa mendapatkan ID instans menggunakan Konsol Manajemen AWS atau perintah describe-instances.
- Pemantauan dasar aktif secara default, tetapi Anda dapat mengaktifkan pemantauan terperinci. Untuk informasi selengkapnya, lihat Kelola pemantauan terperinci untuk instans EC2 Anda.

Untuk menampilkan pemanfaatan CPU pada instans tertentu (konsol)

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, pilih Metrik.
3. Pilih namespace metrik EC2.
4. Pilih dimensi Per-Instance Metrik.
5. Pada bidang pencarian, masukkan **`CPUUtilization`** dan tekan Enter. Pilih baris untuk instans tertentu, yang menampilkan grafik pada metrik CPUUtilization untuk instans tersebut. Untuk memberikan nama pada grafik, pilih ikon pensil. Untuk mengubah rentang waktu, pilih salah satu nilai yang telah ditentukan sebelumnya atau pilih sesuaikan.




6. Untuk mengubah statistik atau periode metrik, pilih tab Metrik grafik. Pilih judul kolom atau nilai individu, lalu pilih nilai yang berbeda.




Untuk mendapatkan pemanfaatan CPU untuk contoh tertentu (AWS CLI)

Gunakan perintah get-metric-statistics berikut untuk mendapatkan metrik CPUUtilization pada instans yang telah ditentukan, menggunakan periode dan interval waktu yang ditentukan:

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name CPUUtilization
 --period 3600 \
--statistics Maximum --dimensions Name=InstanceId,Value=i-1234567890abcdef0 \
--start-time 2022-10-18T23:18:00 --end-time 2022-10-19T23:18:00
```

Berikut contoh outputnya. Setiap nilai merepresentasikan persentase pemanfaatan CPU maksimum untuk satu instans EC2.

```
{
    "Datapoints": [
        {
            "Timestamp": "2022-10-19T00:18:00Z",
            "Maximum": 0.33000000000000002,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-19T03:18:00Z",
            "Maximum": 99.670000000000002,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-19T07:18:00Z",
            "Maximum": 0.34000000000000002,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-19T12:18:00Z",
            "Maximum": 0.34000000000000002,
            "Unit": "Percent"
        }
    ],
    "Label": "CPUUtilization"
}
```

## Mengagregasi statistik di seluruh instans

Statistik agregat tersedia untuk instans yang mengaktifkan pemantauan terperinci. Instans yang menggunakan pemantauan dasar tidak termasuk dalam agregat. Sebelum bisa mendapatkan

statistik agregat di seluruh instans, Anda harus mengaktifkan pemantauan terperinci (dengan biaya tambahan), yang menyediakan data dalam periode 1 menit.

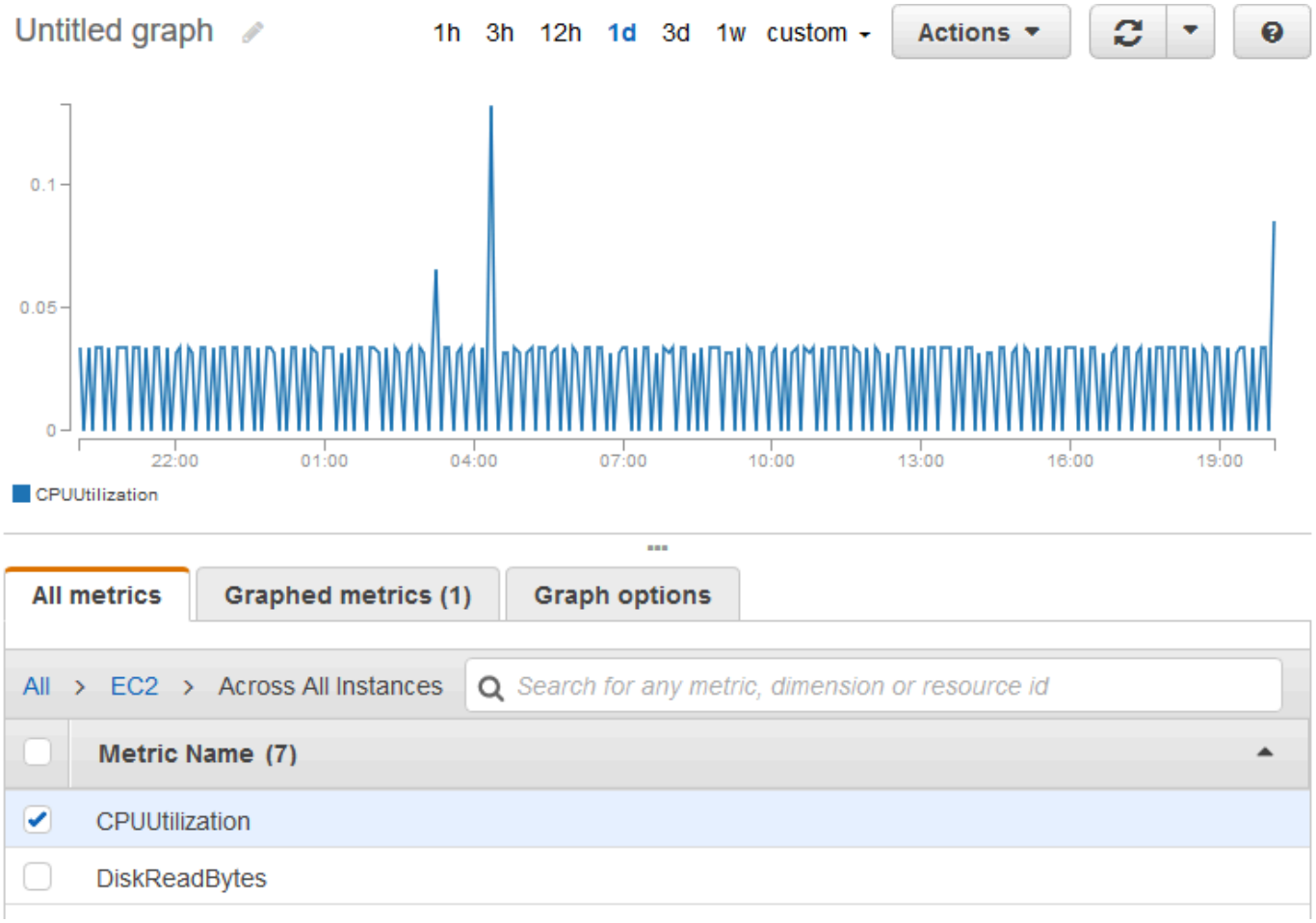
Perhatikan bahwa Amazon CloudWatch tidak dapat mengumpulkan data di seluruh AWS Wilayah. Metrik benar-benar terpisah antar-Wilayah.

Contoh ini menunjukkan cara menggunakan pemantauan terperinci untuk mendapatkan penggunaan CPU rata-rata untuk instans EC2 Anda. Karena tidak ada dimensi yang ditentukan, CloudWatch mengembalikan statistik untuk semua dimensi dalam `AWS/EC2` namespace.

> **Important**
>
> Teknik ini untuk mengambil semua dimensi di seluruh AWS namespace tidak berfungsi untuk ruang nama khusus yang Anda publikasikan ke Amazon. CloudWatch Dengan namespace khusus, Anda harus menentukan rangkaian dimensi lengkap yang terkait dengan titik data mana pun untuk mengambil statistik yang mencakup titik data tersebut.

Untuk menampilkan rata-rata pemanfaatan CPU di seluruh instans Anda (konsol)

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Di panel navigasi, pilih Metrik.
3. Pilih namespace EC2, lalu pilih Di Semua Instans.
4. Pilih baris yang berisi CPUUtilization, yang menampilkan grafik metrik untuk semua instans EC2 Anda. Untuk memberikan nama pada grafik, pilih ikon pensil. Untuk mengubah rentang waktu, pilih salah satu nilai yang telah ditentukan sebelumnya atau pilih sesuaikan.




5. Untuk mengubah statistik atau periode metrik, pilih tab Metrik grafik. Pilih judul kolom atau nilai individu, lalu pilih nilai yang berbeda.

Untuk mendapatkan pemanfaatan CPU rata-rata di seluruh instans Anda (AWS CLI)

Gunakan perintah get-metric-statistics sebagai berikut untuk mendapatkan rata-rata metrik CPUUtilization di seluruh instans Anda.

```
aws cloudwatch get-metric-statistics \
    --namespace AWS/EC2 \
    --metric-name CPUUtilization \
    --period 3600  --statistics "Average" "SampleCount" \
    --start-time 2022-10-11T23:18:00 \
    --end-time 2022-10-12T23:18:00
```

Berikut adalah contoh output:

```
{
    "Datapoints": [
        {
            "SampleCount": 238.0,
            "Timestamp": "2022-10-12T07:18:00Z",
            "Average": 0.038235294117647062,
            "Unit": "Percent"
        },
        {
            "SampleCount": 240.0,
            "Timestamp": "2022-10-12T09:18:00Z",
            "Average": 0.16670833333333332,
            "Unit": "Percent"
        },
        {
            "SampleCount": 238.0,
            "Timestamp": "2022-10-11T23:18:00Z",
            "Average": 0.041596638655462197,
            "Unit": "Percent"
        }
    ],
    "Label": "CPUUtilization"
}
```


## Mengagregasi statistik menurut grup Auto Scaling

Anda dapat mengagregasi statistik untuk instans EC2 dalam grup Auto Scaling. Perhatikan bahwa Amazon CloudWatch tidak dapat mengumpulkan data di seluruh AWS Wilayah. Metrik benar-benar terpisah antar-Wilayah.

Contoh ini menunjukkan cara mengambil total bita yang ditulis ke disk untuk satu grup Auto Scaling. Total tersebut dihitung selama periode 1 menit untuk interval 24 jam di seluruh instans EC2 dalam grup Auto Scaling tertentu.

DiskWriteBytes Untuk menampilkan instans dalam grup Penskalaan Otomatis (konsol)

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Pada panel navigasi, silakan pilih Metrik.
3. Pilih namespace EC2 dan kemudian pilih Berdasarkan grup Auto Scaling.
4. Pilih baris untuk metrik DiskWriteBytes dan grup Auto Scaling tertentu, yang menampilkan grafik untuk metrik pada instans dalam grup Auto Scaling. Untuk memberikan nama pada grafik,

pilih ikon pensil. Untuk mengubah rentang waktu, pilih salah satu nilai yang telah ditentukan sebelumnya atau pilih sesuaikan.

5. Untuk mengubah statistik atau periode metrik, pilih tab Metrik grafik. Pilih judul kolom atau nilai individu, lalu pilih nilai yang berbeda.

DiskWriteBytes Untuk menampilkan instans dalam grup Penskalaan Otomatis (AWS CLI)

Gunakan perintah get-metric-statistics sebagai berikut.

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name DiskWriteBytes
 --period 360 \
--statistics "Sum" "SampleCount" --dimensions Name=AutoScalingGroupName,Value=my-asg --
start-time 2022-10-16T23:18:00 --end-time 2022-10-18T23:18:00
```

Berikut adalah contoh output:

```
{
    "Datapoints": [
        {
            "SampleCount": 18.0,
            "Timestamp": "2022-10-19T21:36:00Z",
            "Sum": 0.0,
            "Unit": "Bytes"
        },
        {
            "SampleCount": 5.0,
            "Timestamp": "2022-10-19T21:42:00Z",
            "Sum": 0.0,
            "Unit": "Bytes"
        }
    ],
    "Label": "DiskWriteBytes"
}
```

## Mengagregasi statistik menurut AMI

Anda dapat mengumpulkan statistik berdasarkan AMI untuk instans yang mengaktifkan pemantauan terperinci. Instans yang menggunakan pemantauan dasar tidak termasuk dalam agregat. Sebelum bisa mendapatkan statistik agregat di seluruh instans, Anda harus mengaktifkan pemantauan terperinci (dengan biaya tambahan), yang menyediakan data dalam periode 1 menit.

Perhatikan bahwa Amazon CloudWatch tidak dapat mengumpulkan data di seluruh AWS Wilayah. Metrik benar-benar terpisah antar-Wilayah.

Contoh ini menunjukkan kepada Anda cara menentukan rata-rata pemanfaatan CPU untuk semua instans yang menggunakan Amazon Machine Image (AMI) tertentu. Rata-rata adalah interval waktu lebih dari 60 detik untuk periode satu hari.

Untuk menampilkan rata-rata pemanfaatan CPU berdasarkan AMI (konsol)

1. Buka CloudWatch konsol di https://console.aws.amazon.com/cloudwatch/.
2. Pada panel navigasi, silakan pilih Metrik.
3. Pilih namespace EC2, lalu pilih Berdasarkan Id Citra (AMI).
4. Pilih baris untuk metrik CPUUtilization dan AMI tertentu, yang menampilkan grafik metrik untuk AMI yang ditentukan. Untuk memberikan nama pada grafik, pilih ikon pensil. Untuk mengubah rentang waktu, pilih salah satu nilai yang telah ditentukan sebelumnya atau pilih sesuaikan.
5. Untuk mengubah statistik atau periode metrik, pilih tab Metrik grafik. Pilih judul kolom atau nilai individu, lalu pilih nilai yang berbeda.

Untuk mendapatkan pemanfaatan CPU rata-rata untuk ID gambar (AWS CLI)

Gunakan perintah get-metric-statistics sebagai berikut.

```
aws cloudwatch get-metric-statistics --namespace AWS/EC2 --metric-name CPUUtilization
 --period 3600 \
--statistics Average --dimensions Name=ImageId,Value=ami-3c47a355 --start-
time 2022-10-10T00:00:00 --end-time 2022-10-11T00:00:00
```

Berikut ini adalah output contoh. Setiap nilai merepresentasikan persentase rata-rata pemanfaatan CPU untuk instans EC2 yang menjalankan AMI yang ditentukan.

```
{
    "Datapoints": [
        {
            "Timestamp": "2022-10-10T07:00:00Z",
            "Average": 0.041000000000000009,
            "Unit": "Percent"
        },
        {
```

```
            "Timestamp": "2022-10-10T14:00:00Z",
            "Average": 0.079579831932773085,
            "Unit": "Percent"
        },
        {
            "Timestamp": "2022-10-10T06:00:00Z",
            "Average": 0.036000000000000011,
            "Unit": "Percent"
        }
    ],
    "Label": "CPUUtilization"
}
```

## Melihat grafik pemantauan untuk instans Anda

Setelah meluncurkan sebuah instans, Anda dapat membuka konsol Amazon EC2 dan melihat grafik pemantauan untuk instans tersebut pada tab Pemantauan. Setiap grafik didasarkan pada salah satu metrik Amazon EC2 yang tersedia.

Berikut adalah grafik yang tersedia:

- Rata-Rata Pemanfaatan CPU (Persen)
- Rata-Rata Pembacaan Disk (Bita)
- Rata-Rata Penulisan Disk (Bita)
- Jaringan Masuk Maksimum (Bita)
- Jaringan Keluar Maksimum (Bita)
- Ringkasan Operasi Baca Disk (Jumlah)
- Ringkasan Operasi Tulis Disk (Jumlah)
- Ringkasan Status (Apa saja)
- Ringkasan Status Instans (Jumlah)
- Ringkasan Status Sistem (Jumlah)

Untuk informasi selengkapnya tentang metrik dan data yang diberikan ke grafik, lihat CloudWatch metrik yang tersedia untuk instans Anda.

Metrik grafik menggunakan CloudWatch konsol

Anda juga dapat menggunakan CloudWatch konsol untuk membuat grafik data metrik yang dihasilkan oleh Amazon EC2 dan AWS layanan lainnya. Untuk informasi selengkapnya, lihat Grafik metrik di Panduan CloudWatch Pengguna Amazon.

## Membuat CloudWatch alarm untuk sebuah instance

Anda dapat membuat CloudWatch alarm yang memantau CloudWatch metrik untuk salah satu instans Anda. CloudWatchakan secara otomatis mengirimi Anda pemberitahuan ketika metrik mencapai ambang batas yang Anda tentukan. Anda dapat membuat CloudWatch alarm menggunakan konsol Amazon EC2, atau menggunakan opsi yang lebih canggih yang disediakan oleh CloudWatch konsol.

Untuk membuat alarm menggunakan CloudWatch konsol

Sebagai contoh, lihat Membuat CloudWatch Alarm Amazon di Panduan CloudWatch Pengguna Amazon.

Untuk membuat alarm menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance dan pilih Tindakan , Monitor dan pemecahan masalah, Kelola CloudWatch alarm.
4. Pada halaman detail CloudWatch Kel ola alarm, di bawah Tambah atau edit alarm, pilih Buat alarm.
5. Untuk Notifikasi alarm, pilih apakah akan mengonfigurasi notifikasi Amazon Simple Notification Service (Amazon SNS). Masukkan topik Amazon SNS yang ada atau masukkan nama untuk membuat topik baru.
6. Untuk Tindakan alarm, pilih apakah akan menentukan tindakan yang akan dilakukan saat alarm dipicu. Pilih tindakan dari dalam daftar.
7. Untuk Ambang batas alarm, pilih metrik dan kriteria untuk alarm. Misalnya, untuk membuat alarm yang dipicu ketika pemanfaatan CPU mencapai 80% selama periode 5 menit, lakukan hal berikut:

    a. Pertahankan pengaturan default untuk Kelompokkan sampel berdasarkan (Rata-rata) dan Tipe data untuk sampel (Pemanfaatan CPU).

    b. Pilih >= untuk Waktu alarm dan masukkan **`0.80`** untuk Persen.

    c. Masukkan **1** untuk periode berturut-turut dan pilih 5 menit untuk Periode.

8. (Opsional) Untuk Data metrik sampel, pilih Tambahkan ke dasbor.
9. Pilih Buat.

Anda dapat mengedit pengaturan CloudWatch alarm dari konsol Amazon EC2 atau CloudWatch konsol. Jika Anda ingin menghapus alarm Anda, Anda dapat melakukannya dari CloudWatch konsol. Untuk informasi selengkapnya, lihat Meng edit atau menghapus CloudWatch alarm di Panduan CloudWatch Pengguna Amazon.

## Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans

Menggunakan tindakan CloudWatch alarm Amazon, Anda dapat membuat alarm yang secara otomatis menghentikan, menghentikan, me-reboot, atau memulihkan instans Anda. Anda dapat menggunakan tindakan penghentian atau pengakhiran untuk membantu menghemat uang saat suatu instans tidak lagi diperlukan. Anda dapat menggunakan tindakan boot ulang dan pemulihan untuk secara otomatis melakukan boot ulang instans tersebut atau memulihkannya ke perangkat keras baru jika terjadi gangguan pada sistem.

> **Note**
>
> Untuk informasi penagihan dan harga CloudWatch alarm Amazon, lihat CloudWatch penagihan dan biaya di Panduan CloudWatch Pengguna Amazon.

Per `AWSServiceRoleForCloudWatchEvents` an terkait layanan memungkinkan AWS untuk melakukan tindakan alarm atas nama Anda. Pertama kali Anda membuat alarm di Konsol Manajemen AWS, AWS CLI, atau IAM API, membuat peran CloudWatch terkait layanan untuk Anda.

Ada sejumlah skenario yang mungkin akan membuat Anda ingin menghentikan atau mengakhiri instans secara otomatis. Misalnya, Anda mungkin memiliki instans khusus untuk membuat batch tugas pemrosesan penggajian atau tugas komputasi ilmiah yang berjalan selama jangka waktu tertentu dan telah menyelesaikan pekerjaannya. Alih-alih membiarkan instans tersebut menganggur (dan menambah biaya), Anda dapat menghentikan atau mengakhirinya, sehingga membantu Anda menghemat uang. Perbedaan utama antara menggunakan tindakan alarm penghentian dan pengakhiran adalah bahwa Anda dapat dengan mudah memulai instans yang dihentikan jika intans tersebut perlu dijalankan kembali nanti. Anda juga dapat menyimpan ID instans dan volume root yang sama. Namun, Anda tidak dapat memulai instans yang diakhiri. Sebaliknya, Anda harus meluncurkan

instans baru. Saat instans dihentikan atau diakhiri, data pada volume penyimpanan instans akan hilang.

Anda dapat menambahkan tindakan hentikan, akhiri, boot ulang, atau pulihkan ke alarm apa pun yang diatur pada metrik Amazon EC2 per-instans, termasuk metrik pemantauan dasar dan terperinci yang disediakan oleh Amazon CloudWatch (di ruang nama `AWS/EC2`), seperti serta metrik khusus apa pun yang menyertakan dimensi `InstanceId`, selama nilainya mengacu pada instans Amazon EC2 yang berjalan dan valid.

> **Important**
>
> Alarm pemeriksaan status dapat memasuki `INSUFFICIENT_DATA` status sementara jika ada titik data metrik yang hilang. Meskipun jarang, ini dapat terjadi ketika ada gangguan dalam sistem pelaporan metrik, bahkan ketika instans sehat. Sebaiknya Anda memperlakukan `INSUFFICIENT_DATA` status sebagai data yang hilang alih-alih pelanggaran alarm, terutama saat mengonfigurasi alarm untuk menghentikan, menghentikan, me-reboot, atau memulihkan instance.

Dukungan konsol

Anda dapat membuat alarm menggunakan konsol Amazon EC2 atau CloudWatch konsol. Prosedur dalam dokumentasi ini menggunakan konsol Amazon EC2. Untuk prosedur yang menggunakan CloudWatch konsol, lihat Membuat alarm yang menghentikan, menghentikan, me-reboot, atau memulihkan instance di Panduan CloudWatch Pengguna Amazon.

Izin

Anda harus memiliki `iam:CreateServiceLinkedRole` untuk membuat atau memodifikasi alarm yang melakukan tindakan alarm EC2. Peran layanan adalah peran IAM yang diasumsikan oleh layanan untuk melakukan tindakan atas nama Anda. Administrator IAM dapat membuat, mengubah, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat Buat sebuah peran untuk mendelegasikan izin ke Layanan AWS dalam Panduan pengguna IAM.

Daftar Isi


- Tambahkan tindakan berhenti ke CloudWatch alarm Amazon
- Tambahkan tindakan penghentian ke CloudWatch alarm Amazon
- Tambahkan tindakan reboot ke CloudWatch alarm Amazon

- Tambahkan tindakan pemulihan ke CloudWatch alarm Amazon
- Skenario tindakan CloudWatch alarm Amazon

## Tambahkan tindakan berhenti ke CloudWatch alarm Amazon

Anda dapat membuat alarm yang menghentikan instans Amazon EC2 ketika ambang batas tertentu telah terpenuhi. Misalnya, Anda dapat mengoperasikan pengembangan atau instans pengujian dan terkadang lupa untuk mematikannya. Anda dapat membuat alarm yang dipicu ketika persentase rata-rata pemanfaatan CPU kurang dari 10 persen selama 24 jam, yang menandakan bahwa alarm dalam keadaan menganggur dan tidak diperlukan lagi. Anda dapat menyesuaikan ambang batas, durasi, dan periode sesuai kebutuhan. Anda juga dapat menambahkan notifikasi Amazon Simple Notification Service (Amazon SNS) untuk menerima email saat alarm dipicu.

Instans yang menggunakan volume Amazon EBS sebagai volume root dapat dihentikan atau dihentikan, sedangkan instans yang menggunakan penyimpanan instans sebagai volume root hanya dapat dihentikan. Data pada volume penyimpanan instans hilang saat instans diakhiri atau dihentikan.

Untuk membuat alarm agar dapat menghentikan instans yang menganggur (konsol Amazon EC2)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance dan pilih Tindakan , Monitor dan pemecahan masalah, Kelola CloudWatch alarm.

   Atau, Anda dapat memilih tanda plus (

   ) pada kolom Status alarm.
4. Pada halaman Kelola CloudWatch alarm, lakukan hal berikut:

   a. Pilih Buat alarm.

   b. Untuk menerima email saat alarm dipicu, untuk Notifikasi alarm, pilih topik Amazon SNS yang ada. Anda harus terlebih dahulu membuat topik Amazon SNS menggunakan konsol Amazon SNS. Untuk informasi selengkapnya, lihat Menggunakan Amazon SNS untuk perpesanan aplikasi-ke-orang (A2P) dalam Panduan Developer Amazon Simple Notification Service.

   c. Aktifkan Tindakan alarm, lalu pilih Hentikan.

d. Untuk Kelompokkan sampel berdasarkan serta Tipe data untuk sampel, pilih statistik dan metrik. Dalam contoh ini, pilih Rata-rata dan Pemanfaatan CPU.

e. Untuk Waktu Alarm dan Persen, tentukan ambang batas metrik. Dalam contoh ini, pilih <= dan 10 persen.

f. Untuk Periode berturut-turut dan Periode, tentukan periode evaluasi alarm. Dalam contoh ini, pilih 1 periode berturut-turut 5 Menit.

g. Amazon CloudWatch secara otomatis membuat nama alarm untuk Anda. Untuk mengganti nama, pada Nama alarm, masukkan nama baru. Nama alarm hanya boleh berisi karakter ASCII.

> **Note**
>
> Anda dapat menyesuaikan konfigurasi alarm berdasarkan kebutuhan sebelum membuat alarm, atau Anda dapat mengeditnya nanti. Penyesuaian ini termasuk pengaturan metrik, ambang batas, durasi, tindakan, dan notifikasi. Namun, nama alarm yang telah dibuat sudah tidak dapat diedit.

h. Pilih Buat.

## Tambahkan tindakan penghentian ke CloudWatch alarm Amazon

Anda dapat membuat alarm yang mengakhiri instans EC2 secara otomatis ketika ambang batas tertentu telah terpenuhi (selama perlindungan pengakhiran tidak diaktifkan untuk instans tersebut). Misalnya, Anda mungkin ingin mengakhiri instans ketika telah menyelesaikan pekerjaannya dan sudah tidak diperlukan lagi. Jika Anda mungkin ingin menggunakan instans tersebut nanti, Anda sebaiknya menghentikan instans tersebut dan tidak menghentikannya. Data pada volume penyimpanan instans hilang saat instans diakhiri. Untuk informasi tentang pengaktifan dan penonaktifan perlindungan pengakhiran pada instans, lihat Ubah perlindungan penghentian instans.

Untuk membuat alarm agar dapat mengakhiri instans yang menganggur (konsol Amazon EC2)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance dan pilih Tindakan , Monitor dan pemecahan masalah, Kelola CloudWatch alarm.

Atau, Anda dapat memilih tanda plus (



) pada kolom Status alarm.

4. Pada halaman Kelola CloudWatch alarm, lakukan hal berikut:

   a. Pilih Buat alarm.

   b. Untuk menerima email saat alarm dipicu, untuk Notifikasi alarm, pilih topik Amazon SNS yang ada. Anda harus terlebih dahulu membuat topik Amazon SNS menggunakan konsol Amazon SNS. Untuk informasi selengkapnya, lihat Menggunakan Amazon SNS untuk perpesanan aplikasi-ke-orang (A2P) dalam Panduan Developer Amazon Simple Notification Service.

   c. Aktifkan Tindakan alarm, lalu pilih Akhiri.

   d. Untuk Kelompokkan sampel berdasarkan serta Tipe data untuk sampel, pilih statistik dan metrik. Dalam contoh ini, pilih Rata-rata dan Pemanfaatan CPU.

   e. Untuk Waktu Alarm dan Persen, tentukan ambang batas metrik. Dalam contoh ini, pilih => dan 10 persen.

   f. Untuk Periode berturut-turut dan Periode, tentukan periode evaluasi alarm. Dalam contoh ini, pilih 24 periode berturut-turut dari 1 Jam.

   g. Amazon CloudWatch secara otomatis membuat nama alarm untuk Anda. Untuk mengganti nama, pada Nama alarm, masukkan nama baru. Nama alarm hanya boleh berisi karakter ASCII.

      > **Note**
      >
      > Anda dapat menyesuaikan konfigurasi alarm berdasarkan kebutuhan sebelum membuat alarm, atau Anda dapat mengeditnya nanti. Penyesuaian ini termasuk pengaturan metrik, ambang batas, durasi, tindakan, dan notifikasi. Namun, nama alarm yang telah dibuat sudah tidak dapat diedit.

   h. Pilih Buat.

## Tambahkan tindakan reboot ke CloudWatch alarm Amazon

Anda dapat membuat CloudWatch alarm Amazon yang memantau instans Amazon EC2 dan secara otomatis me-reboot instans. Tindakan alarm boot ulang direkomendasikan untuk kegagalan

Pemeriksaan Kondisi instans (sebagai lawan dari tindakan alarm pemulihan, yang sesuai untuk kegagalan Pemeriksaan Kondisi Sistem). Sebuah instans yang melakukan boot ulang setara dengan penyalaan ulang sistem operasi. Dalam kebanyakan kasus, hanya diperlukan beberapa menit untuk menyalakan ulang instans Anda. Saat Anda melakukan boot ulang, instans tetap berada di host fisik yang sama, sehingga instans Anda tetap menggunakan nama DNS publik, alamat IP privat, dan setiap data pada volume penyimpanan instansnya.

Boot ulang instans tidak memulai periode penagihan instans baru (dengan biaya minimum satu menit), tidak seperti penghentian dan pemulaian ulang instans Anda. Data pada volume penyimpanan instans dipertahankan saat instans di-boot ulang. Volume penyimpanan instans harus dipasang kembali ke sistem file setelah boot ulang. Untuk informasi selengkapnya, lihat Mulai ulang instans Amazon EC2 Anda.

> **Important**
>
> Untuk menghindari kondisi pacu antara tindakan boot ulang dan pemulihan, jangan mengatur jumlah periode evaluasi yang sama untuk alarm boot ulang dan alarm pemulihan. Kami menyarankan Anda untuk mengatur alarm boot ulang ke tiga periode evaluasi, masing-masing selama satu menit. Untuk informasi selengkapnya, lihat Meng evaluasi alarm di Panduan CloudWatch Pengguna Amazon.

Untuk membuat alarm agar dapat melakukan boot ulang instans (konsol Amazon EC2)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance dan pilih Tindakan , Monitor dan pemecahan masalah, Kelola CloudWatch alarm.

   Atau, Anda dapat memilih tanda plus (

   ) pada kolom Status alarm.
4. Pada halaman Kelola CloudWatch alarm, lakukan hal berikut:

   a. Pilih Buat alarm.

   b. Untuk menerima email saat alarm dipicu, untuk Notifikasi alarm, pilih topik Amazon SNS yang ada. Anda harus terlebih dahulu membuat topik Amazon SNS menggunakan konsol Amazon SNS. Untuk informasi selengkapnya, lihat Menggunakan Amazon SNS untuk

perpesanan aplikasi-ke-orang (A2P) dalam Panduan Developer Amazon Simple Notification Service.

c. Aktifkan Tindakan alarm, lalu pilih Boot ulang.

d. Untuk Kelompokkan sampel berdasarkan serta Tipe data untuk sampel, pilih statistik dan metrik. Pada contoh ini, pilih Rata-rata dan Pemeriksaan status gagal: instans.

e. Untuk Periode berturut-turut dan Periode, tentukan periode evaluasi alarm. Dalam contoh ini, masukkan 3 periode berturut-turut 1 Menit. Jika 1 Menit dinonaktifkan, Anda harus mengaktifkan pemantauan terperinci, atau Anda dapat memilih 5 Menit sebagai gantinya.

f. Amazon CloudWatch secara otomatis membuat nama alarm untuk Anda. Untuk mengganti nama, pada Nama alarm, masukkan nama baru. Nama alarm hanya boleh berisi karakter ASCII.

g. Pilih Buat.

## Tambahkan tindakan pemulihan ke CloudWatch alarm Amazon

Anda dapat membuat CloudWatch alarm Amazon yang memantau instans Amazon EC2. Jika instans menjadi rusak karena kegagalan perangkat keras yang mendasarinya atau masalah yang memerlukan AWS keterlibatan untuk diperbaiki, Anda dapat memulihkan instans secara otomatis. Instans yang diakhiri tidak dapat dipulihkan. Instans yang dipulihkan identik dengan instans awal, termasuk ID instans , alamat IP privat, alamat IP Elastis, dan semua metadata instans.

CloudWatch mencegah Anda menambahkan tindakan pemulihan ke alarm yang ada pada instance yang tidak mendukung tindakan pemulihan.

Ketika alarm `StatusCheckFailed_System` dipicu dan tindakan pemulihan dimulai, topik Amazon SNS yang Anda pilih akan mengirimkan notifikasi ketika Anda membuat alarm dan mengaitkan tindakan pemulihan. Selama pemulihan, instans dimigrasikan selama boot ulang instans, dan semua data yang berada dalam memori akan hilang. Saat proses selesai, informasi akan diterbitkan ke topik SNS yang telah Anda konfigurasikan untuk alarm. Setiap orang yang berlangganan topik SNS ini akan menerima notifikasi email yang meliputi status upaya pemulihan dan petunjuk lebih lanjut. Anda melihat boot ulang instans pada instans yang dipulihkan.

> **Note**
>
> Tindakan pemulihan hanya dapat digunakan dengan `StatusCheckFailed_System`, tidak dengan `StatusCheckFailed_Instance`.

Masalah berikut dapat menyebabkan kegagalan pemeriksaan status sistem:

- Hilangnya konektivitas jaringan
- Kehilangan daya sistem
- Masalah perangkat lunak pada host fisik
- Masalah perangkat keras pada host fisik yang memengaruhi jangkauan jaringan

Tindakan pemulihan hanya didukung pada instans yang memenuhi karakteristik tertentu. Untuk informasi selengkapnya, lihat Pemulihan instans otomatis.

Jika instans Anda memiliki alamat IP publik, instans tersebut akan mempertahankan alamat IP publik setelah pemulihan.

> **Important**
>
> Untuk menghindari kondisi pacu antara tindakan boot ulang dan pemulihan, jangan mengatur jumlah periode evaluasi yang sama untuk alarm boot ulang dan alarm pemulihan. Kami menyarankan Anda untuk mengatur alarm pemulihan ke dua periode evaluasi, masing-masing selama satu menit. Untuk informasi selengkapnya, lihat Meng evaluasi alarm di Panduan CloudWatch Pengguna Amazon.

Untuk membuat alarm agar dapat memulihkan instans (konsol Amazon EC2)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instance dan pilih Tindakan , Monitor dan pemecahan masalah, Kelola CloudWatch alarm.

   Atau, Anda dapat memilih tanda plus (

   ) pada kolom Status alarm.
4. Pada halaman Kelola CloudWatch alarm, lakukan hal berikut:

   a. Pilih Buat alarm.

   b. Untuk menerima email saat alarm dipicu, untuk Notifikasi alarm, pilih topik Amazon SNS yang ada. Anda harus terlebih dahulu membuat topik Amazon SNS menggunakan konsol

Amazon SNS. Untuk informasi selengkapnya, lihat Menggunakan Amazon SNS untuk perpesanan aplikasi-ke-orang (A2P) dalam Panduan Developer Amazon Simple Notification Service.

> **Note**
>
> Pengguna harus berlangganan topik SNS yang ditentukan untuk menerima notifikasi email saat alarm dipicu. Sel Pengguna root akun AWS alu menerima pemberitahuan email saat tindakan pemulihan instans otomatis terjadi, bahkan jika topik SNS tidak ditentukan atau pengguna root tidak berlangganan topik SNS yang ditentukan.

c. Aktifkan Tindakan alarm, lalu pilih Pulihkan.

d. Untuk Kelompokkan sampel berdasarkan serta Tipe data untuk sampel, pilih statistik dan metrik. Pada contoh ini, pilih Rata-rata dan Pemeriksaan status gagal: sistem.

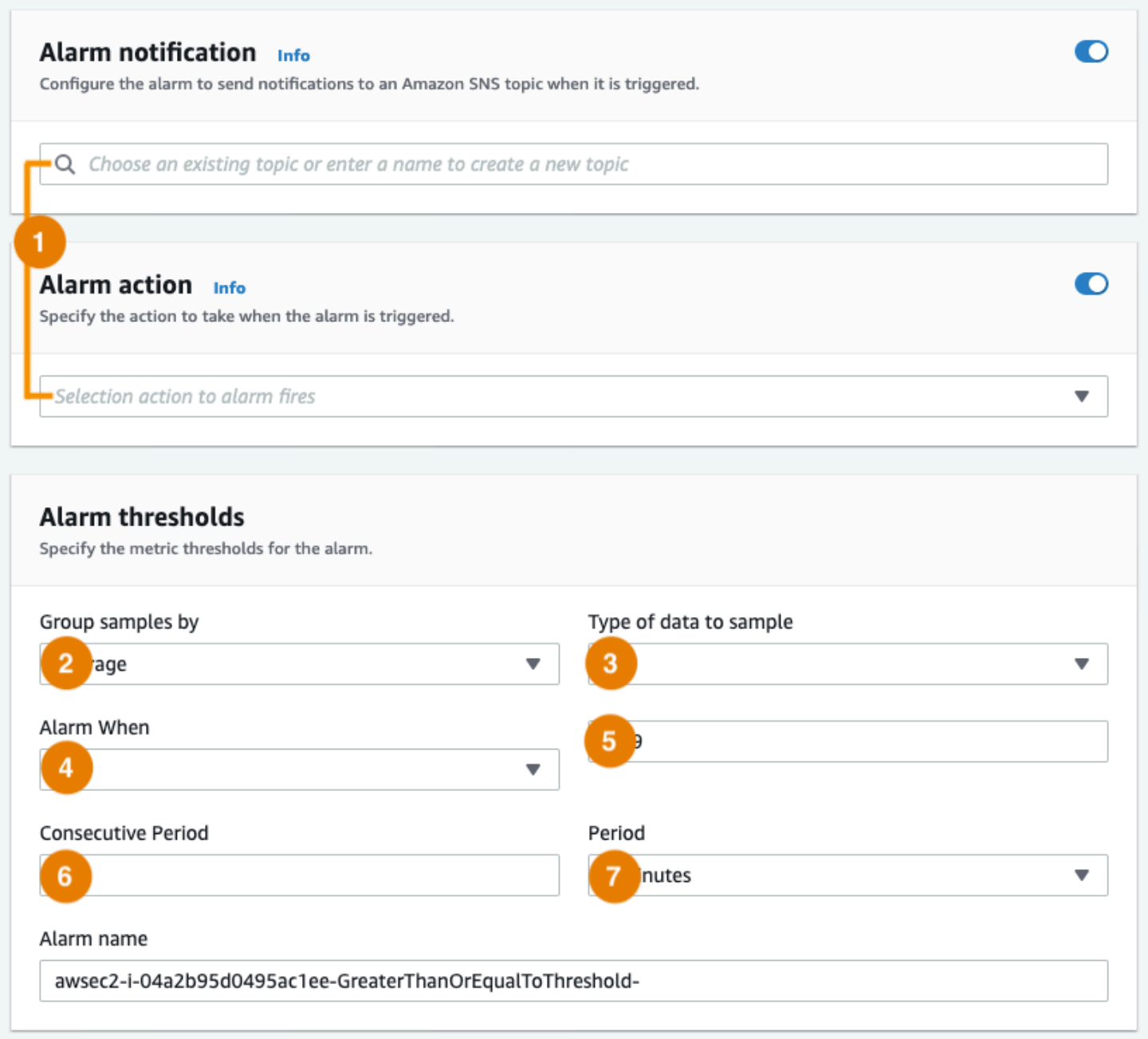
e. Untuk Periode berturut-turut dan Periode, tentukan periode evaluasi alarm. Dalam contoh ini, masukkan 2 periode berturut-turut 1 Menit. Jika 1 Menit dinonaktifkan, Anda harus mengaktifkan pemantauan terperinci, atau Anda dapat memilih 5 Menit sebagai gantinya.

f. Amazon CloudWatch secara otomatis membuat nama alarm untuk Anda. Untuk mengganti nama, pada Nama alarm, masukkan nama baru. Nama alarm hanya boleh berisi karakter ASCII.

g. Pilih Buat.

## Skenario tindakan CloudWatch alarm Amazon

Anda dapat menggunakan konsol Amazon EC2 untuk membuat tindakan alarm yang menghentikan atau mengakhiri instans Amazon EC2 ketika kondisi tertentu terpenuhi. Pada tangkapan layar halaman konsol tempat Anda mengatur tindakan alarm berikut, kami telah menomori pengaturannya. Kami juga telah menomori pengaturan dalam skenario yang mengikuti untuk membantu Anda membuat tindakan yang tepat.




Skenario 1: Menghentikan instans pengembangan dan pengujian yang menganggur

Buat alarm yang menghentikan instans yang digunakan untuk pengembangan atau pengujian perangkat lunak saat sedang menganggur selama setidaknya satu jam.

| Pengaturan | Nilai |
| --- | --- |
| 1 | Berhenti |

| Pengaturan | Nilai |
| --- | --- |
| 2 | Maksimum |
| 3 | Pemanfaatan CPU |
| 4 | <= |
| 5 | 10% |
| 6 | 1 |
| 7 | 1 Jam |

Skenario 2: Menghentikan instans yang menganggur

Buat alarm yang menghentikan sebuah instans dan mengirimkan email saat instans tersebut sudah menganggur selama 24 jam.

| Pengaturan | Nilai |
| --- | --- |
| 1 | Hentikan dan kirim email |
| 2 | Rata-rata |
| 3 | Pemanfaatan CPU |
| 4 | <= |
| 5 | 5% |
| 6 | 24 |
| 7 | 1 Jam |

Skenario 3: Mengirimkan email mengenai server web dengan lalu lintas yang luar biasa tinggi

Buat alarm yang mengirimkan email ketika sebuah instans melebihi 10 GB lalu lintas jaringan keluar per hari.

| Pengaturan | Nilai |
| --- | --- |
| 1 | Email |
| 2 | Jumlah |
| 3 | Jaringan Keluar |
| 4 | > |
| 5 | 10 GB |
| 6 | 24 |
| 7 | 1 Jam |

Skenario 4: Menghentikan server web dengan lalu lintas yang luar biasa tinggi

Buat alarm yang menghentikan instans dan kirim pesan teks (SMS) jika lalu lintas keluar melebihi 1 GB per jam.

| Pengaturan | Nilai |
| --- | --- |
| 1 | Hentikan dan kirim SMS |
| 2 | Jumlah |
| 3 | Jaringan Keluar |
| 4 | > |
| 5 | 1 GB |
| 6 | 1 |
| 7 | 1 Jam |

Skenario 5: Menghentikan instans yang terganggu

Buat alarm yang menghentikan instans yang gagal dalam tiga pemeriksaan status berturut-turut (dilakukan dengan interval 5 menit).

| Pengaturan | Nilai |
|---|---|
| 1 | Berhenti |
| 2 | Rata-rata |
| 3 | Pemeriksaan Status Gagal: (Sistem) |
| 4 | - |
| 5 | - |
| 6 | 1 |
| 7 | 15 Menit |

Skenario 6: Mengakhiri instans ketika pembuatan batch pekerjaan pemrosesan selesai

Buat alarm yang mengakhiri instans yang menjalankan tugas batch jika tidak lagi mengirimkan data hasil.

| Pengaturan | Nilai |
|---|---|
| 1 | Akhiri |
| 2 | Maksimum |
| 3 | Jaringan Keluar |
| 4 | <= |
| 5 | 100.000 bita |
| 6 | 1 |

| Pengaturan | Nilai |
| --- | --- |
| 7 | 5 Menit |

# Otomatiskan Amazon menggunakan EC2 EventBridge

Anda dapat menggunakan Amazon EventBridge untuk mengotomatiskan Layanan AWS dan merespons peristiwa sistem secara otomatis, seperti masalah ketersediaan aplikasi atau perubahan sumber daya. Acara dari AWS layanan dikirimkan ke EventBridge dalam waktu dekat. Anda dapat membuat aturan untuk menunjukkan peristiwa yang sesuai kepentingan Anda, dan tindakan yang akan diambil ketika peristiwa sesuai dengan aturan. Tindakan yang dapat dipicu secara otomatis meliputi hal-hal berikut:

- Memanggil fungsi AWS Lambda
- Memanggil Amazon EC2 Run Command
- Menyampaikan peristiwa ke Amazon Kinesis Data Streams
- Aktifkan mesin AWS Step Functions negara
- Mengirim notifikasi topik Amazon SNS
- Mengirim notifikasi antrean Amazon SQS

Berikut ini adalah contoh bagaimana Anda dapat menggunakan EventBridge dengan Amazon EC2:

- Aktifkan fungsi Lambda setiap kali instans memasuki status berjalan.
- Notifikasi topik Amazon SNS saat volume Amazon EBS dibuat atau dimodifikasi.
- Kirim perintah ke satu atau beberapa EC2 instans Amazon menggunakan Amazon EC2 Run Command setiap kali peristiwa tertentu di AWS layanan lain terjadi.

Untuk informasi selengkapnya, lihat Panduan EventBridge Pengguna Amazon.

## Jenis EC2 acara Amazon

Amazon EC2 mendukung jenis acara berikut:

- EC2 Perubahan Negara AMI
- EC2 Pemberitahuan Perubahan Status Peluncuran Cepat

- EC2 Kesalahan Armada
- EC2 Informasi Armada
- EC2 Perubahan Instans Armada
- EC2 Perubahan Permintaan Instans Fleet Spot
- EC2 Perubahan Negara Armada
- EC2 Rekomendasi Penyeimbangan Ulang Instance
- EC2 Pemberitahuan Perubahan Status Instance
- EC2 Kesalahan Armada Spot
- EC2 Informasi Armada Spot
- EC2 Perubahan Instans Armada Spot
- EC2 Perubahan Permintaan Instans Spot Armada Spot
- EC2 Perubahan Negara Armada Spot
- EC2 Peringatan Gangguan Instans Spot
- EC2 Pemenuhan Permintaan Instans Spot
- EC2 Pemberitahuan Kurangnya Pemanfaatan ODCR

Untuk informasi tentang jenis acara yang didukung oleh Amazon EBS, lihat Amazon EventBridge untuk Amazon EBS.

## Log panggilan API Amazon EC2 menggunakan AWS CloudTrail

Amazon EC2 API terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau. Layanan AWS CloudTrail menangkap semua panggilan API Amazon EC2 sebagai peristiwa. Panggilan yang ditangkap termasuk panggilan yang dilakukan oleh konsol. Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat ke Amazon EC2 API, alamat IP dari mana permintaan dibuat, dan kapan permintaan itu dibuat.

Setiap entri peristiwa atau log berisi informasi tentang entitas yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan berikut hal ini:

- Baik permintaan tersebut dibuat dengan kredensial pengguna root atau pengguna.

- Apakah permintaan dibuat atas nama pengguna Pusat Identitas IAM.
- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara untuk satu peran atau pengguna gabungan.
- Apakah permintaan tersebut dibuat oleh Layanan AWS lain.

CloudTrail aktif di akun Anda Akun AWS saat Anda membuat akun dan Anda secara otomatis memiliki akses ke riwayat CloudTrail Acara. Ri CloudTrail wayat acara menyediakan catatan yang dapat dilihat, dapat dicari, dapat diunduh, dan tidak dapat diubah dari 90 hari terakhir dari peristiwa manajemen yang direkam dalam sebuah. Wilayah AWS Untuk informasi selengkapnya, lihat Bek erja dengan riwayat CloudTrail acara di Panduan AWS CloudTrail Pengguna. Tidak ada CloudTrail biaya untuk melihat ri wayat acara.

Untuk catatan peristiwa yang berkelanjutan dalam 90 hari Akun AWS terakhir Anda, buat jejak atau penyimpanan data acara CloudTrail Lake.

CloudTrail jalan setapak

Jej ak CloudTrail memungkinkan pengiriman file log ke bucket Amazon S3. Semua jalur yang dibuat menggunakan Multi-wilayah. Konsol Manajemen AWS Anda dapat membuat jejak wilayah tunggal atau multi-wilayah dengan menggunakan. AWS CLI Membuat jejak Multi-wilayah disarankan karena Anda menangkap aktivitas di semua Wilayah AWS di akun Anda. Jika Anda membuat jejak satu wilayah, Anda hanya dapat melihat peristiwa yang dicatat di jalur tersebut Wilayah AWS. Untuk informasi selengkapnya tentang jalur, lihat Membuat jejak untuk Anda Akun AWS dan Membuat jejak untuk organisasi di Panduan AWS CloudTrail Pengguna.

Anda dapat mengirimkan satu salinan acara manajemen yang sedang berlangsung ke bucket Amazon S3 Anda tanpa biaya CloudTrail dengan membuat jejak, namun, ada biaya penyimpanan Amazon S3. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga. Untuk informasi tentang harga Amazon S3, lihat Harga Amazon S3.

CloudTrail Penyimpanan data acara danau

CloudTrail Lake memungkinkan Anda menjalankan SQL-based kueri pada acara Anda. CloudTrail Lake mengonversi peristiwa yang ada dalam format JSON berbasis baris ke format Apache ORC. ORC adalah format penyimpanan kolumnar yang dioptimalkan untuk pengambilan data dengan cepat. Peristiwa digabungkan ke dalam penyimpanan data peristiwa, yang merupakan kumpulan peristiwa yang tidak dapat diubah berdasarkan kriteria yang Anda pilih dengan menerapkan pemilih peristiwa lanj utan. Pemilih yang Anda terapkan ke penyimpanan data peristiwa mengontrol peristiwa mana yang bertahan dan tersedia untuk Anda kueri. Untuk

informasi selengkapnya tentang CloudTrail Danau, lihat Bek erja dengan AWS CloudTrail Danau di Panduan AWS CloudTrail Pengguna.

CloudTrail Penyimpanan dan kueri data acara danau dikenakan biaya. Saat membuat penyimpanan data peristiwa, Anda memilih opsi harga yang ingin Anda gunakan untuk penyimpanan data peristiwa. Opsi penetapan harga menentukan biaya untuk menelan dan menyimpan peristiwa, serta periode retensi default dan maksimum untuk penyimpanan data peristiwa. Untuk informasi selengkapnya tentang CloudTrail harga, lihat AWS CloudTrail Harga.

## Acara manajemen API Amazon EC2 di CloudTrail

Acara manajemen memberikan informasi tentang operasi manajemen yang dilakukan pada sumber daya di Anda Akun AWS. Ini juga dikenal sebagai operasi bidang kontrol. Secara default, CloudTrail mencatat peristiwa manajemen.

Semua tindakan Amazon EC2 API dicatat sebagai peristiwa manajemen. Untuk daftar tindakan API yang dicatat CloudTrail, lihat Refer ensi API Amazon EC2. Misalnya, panggilan ke RunInstances, DescribeInstances, dan StopInstances tindakan dicatat sebagai peristiwa manajemen.

## Contoh acara API Amazon EC2

Peristiwa mewakili permintaan tunggal dari sumber apa pun dan mencakup informasi tentang operasi API yang diminta, tanggal dan waktu operasi, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi acara tidak muncul dalam urutan tertentu.

Catatan file log berikut menunjukkan bahwa pengguna telah mengakhiri sebuah instans.


```
{
   "Records":[
      {
         "eventVersion":"1.03",
         "userIdentity":{
            "type":"Root",
            "principalId":"123456789012",
            "arn":"arn:aws:iam::123456789012:root",
            "accountId":"123456789012",
            "accessKeyId":"AKIAIOSFODNN7EXAMPLE",
            "userName":"user"
         },
```

```
        "eventTime":"2016-05-20T08:27:45Z",
        "eventSource":"ec2.amazonaws.com",
        "eventName":"TerminateInstances",
        "awsRegion":"us-west-2",
        "sourceIPAddress":"198.51.100.1",
        "userAgent":"aws-cli/1.10.10 Python/2.7.9 Windows/7botocore/1.4.1",
        "requestParameters":{
           "instancesSet":{
              "items":[{
                 "instanceId":"i-1a2b3c4d"
              }]
           }
        },
        "responseElements":{
           "instancesSet":{
              "items":[{
                 "instanceId":"i-1a2b3c4d",
                 "currentState":{
                    "code":32,
                    "name":"shutting-down"
                 },
                 "previousState":{
                    "code":16,
                    "name":"running"
                 }
              }]
           }
        },
        "requestID":"be112233-1ba5-4ae0-8e2b-1c302EXAMPLE",
        "eventID":"6e12345-2a4e-417c-aa78-7594fEXAMPLE",
        "eventType":"AwsApiCall",
        "recipientAccountId":"123456789012"
     }
   ]
}
```


Untuk informasi tentang konten CloudTrail rekaman, lihat konten CloudTrail rek aman di Panduan AWS CloudTrail Pengguna.

## Koneksi audit yang dibuat menggunakan EC2 Instance Connect

Anda dapat menggunakannya AWS CloudTrail untuk mengaudit pengguna yang terhubung ke instans menggunakan EC2 Instance Connect.

Untuk mengaudit aktivitas SSH melalui EC2 Instance Connect menggunakan AWS CloudTrail konsol

1. Buka CloudTrail konsol di https://console.aws.amazon.com/cloudtrail/.
2. Pastikan Anda berada di Wilayah yang benar.
3. Di panel navigasi, pilih Riwayat Peristiwa.
4. Untuk Filter, pilih Sumber peristiwa, ec2-instance-connect.amazonaws.com.
5. (Opsional) Untuk Rentang waktu, pilih satu rentang waktu.
6. Pilih ikon Segarkan peristiwa.
7. Halaman ini menampilkan peristiwa yang sesuai dengan panggilan API SendSSHPublicKey. Perluas acara menggunakan panah untuk melihat detail tambahan, seperti nama pengguna dan kunci AWS akses yang digunakan untuk membuat koneksi SSH, dan alamat IP sumber.
8. Untuk menampilkan informasi peristiwa yang lengkap dalam format JSON, pilih Lihat peristiwa. Bid ang requestParam eters berisi ID instance tujuan, nama pengguna OS, dan kunci publik yang digunakan untuk membuat koneksi SSH.


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDEFGONGNOMOOCB6XYTQEXAMPLE",
        "arn": "arn:aws:iam::1234567890120:user/IAM-friendly-name",
        "accountId": "123456789012",
        "accessKeyId": "ABCDEFGUKZHNAW4OSN2AEXAMPLE",
        "userName": "IAM-friendly-name",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-09-21T21:37:58Z"}
        }
    },
    "eventTime": "2018-09-21T21:38:00Z",
    "eventSource": "ec2-instance-connect.amazonaws.com",
    "eventName": "SendSSHPublicKey ",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "123.456.789.012",
    "userAgent": "aws-cli/1.15.61 Python/2.7.10 Darwin/16.7.0 botocore/1.10.60",
    "requestParameters": {
        "instanceId": "i-0123456789EXAMPLE",
        "osUser": "ec2-user",
        "SSHKey": {
```

```
            "publicKey": "ssh-rsa ABCDEFGHIJKLMNO01234567890EXAMPLE"
        }
     },
    "responseElements": null,
    "requestID": "1a2s3d4f-bde6-11e8-a892-f7ec64543add",
    "eventID": "1a2w3d4r5-a88f-4e28-b3bf-30161f75be34",
    "eventType": "AwsApiCall",
    "recipientAccountId": "0987654321"
}
```


Jika Anda telah mengonfigurasi AWS akun Anda untuk mengumpulkan CloudTrail peristiwa dalam bucket S3, Anda dapat mengunduh dan mengaudit informasi secara terprogram. Untuk informasi selengkapnya, lihat Mendapatkan dan melihat file CloudTrail log Anda di Panduan AWS CloudTrail Pengguna.

# Memantau aplikasi.NET dan SQL Server menggunakan Appl CloudWatch ication Insights

CloudWatch Application Insights membantu Anda memantau aplikasi.NET dan SQL Server yang menggunakan instans Amazon EC2 bersama dengan sumber daya AWS aplikasi lainnya. Ini mengidentifikasi dan mengatur log metrik utama dan alarm di seluruh sumber daya aplikasi dan tumpukan teknologi Anda (misalnya, database Microsoft SQL Server Anda, server web (IIS) dan aplikasi, OS, penyeimbang beban, dan antrian). Wawasan Aplikasi CloudWatch terus memantau metrik dan log untuk mendeteksi serta menghubungkan anomali dan kesalahan. Ketika kesalahan dan anomali terdeteksi, Application Insights menghasilkan peristiwa yang dapat Anda gunakan untuk mengatur notifikasi atau mengambil tindakan. Untuk membantu memecahkan masalah, Wawasan Aplikasi membuat dasbor otomatis pada masalah yang terdeteksi, yang mencakup anomali metrik dan kesalahan log yang berhubungan, beserta wawasan tambahan untuk menunjukkan kemungkinan akar masalah. Dasbor otomatis tersebut membantu Anda mengambil tindakan perbaikan untuk menjaga aplikasi agar tetap sehat dan mencegah dampak bagi pengguna akhir aplikasi Anda.

Informasi yang diberikan terkait masalah yang terdeteksi:

- Ringkasan singkat masalah
- Waktu mulai dan tanggal masalah
- Tingkat keparahan masalah: High/Medium/Low
- Status masalah yang terdeteksi: Dalam progress/Resolved

- Wawasan: Secara otomatis menghasilkan wawasan terkait masalah yang terdeteksi dan kemungkinan akar masalah
- Umpan balik tentang wawasan: Umpan balik yang telah Anda berikan tentang kegunaan wawasan yang dihasilkan oleh CloudWatch Application Insights untuk.NET dan SQL Server
- Observasi terkait: Tampilan terperinci dari anomali metrik dan cuplikan kesalahan dari log yang relevan terkait masalah di berbagai komponen aplikasi

Umpan Balik

Anda dapat memberikan umpan balik mengenai wawasan yang dihasilkan secara otomatis terkait masalah yang terdeteksi dengan menetapkannya sebagai berguna atau tidak berguna. Umpan balik mengenai wawasan tersebut, beserta diagnostik aplikasi Anda (anomali metrik dan pengecualian log), digunakan untuk meningkatkan deteksi masalah serupa pada masa mendatang.

Untuk informasi selengkapnya, lihat dokumentasi Appl CloudWatch ication Insights di Panduan CloudWatch Pengguna Amazon.

# Lacak penggunaan Tingkat Gratis Anda untuk Amazon EC2

Saat Anda membuat Akun AWS, Anda dapat memulai dengan Amazon EC2 secara gratis menggunakan Tingkat AWS Gratis. Manfaat Tingkat Gratis Anda berbeda tergantung pada apakah Anda membuat akun sebelum 15 Juli 2025, atau pada atau setelah 15 Juli 2025. Untuk informasi selengkapnya, lihat Jel AWS ajahi layanan dengan Tingkat AWS Gratis di Panduan AWS Billing Pengguna.

## Manfaat Tingkat Gratis sebelum dan sesudah 15 Juli 2025

Tabel berikut membandingkan manfaat Tingkat Gratis berdasarkan tanggal Akun AWS pembuatan Anda:

| Manfaat Tingkat Gratis | Akun dibuat sebelum 15 Juli 2025 | Akun dibuat pada atau setelah 15 Juli 2025 |
|---|---|---|
| Jenis instans yang ditandai Tingkat gratis memenuhi syarat | `t2.micro`, `t3.micro` | `t3.micro`, `t3.small`, `t4g.micro` , `t4g.small` |

| Manfaat Tingkat Gratis | Akun dibuat sebelum 15 Juli 2025 | Akun dibuat pada atau setelah 15 Juli 2025 |
|---|---|---|
| | | , `c7i-flex.large` , `m7i-flex.large` |
| Jenis volume Amazon EBS ditandai Tingkat gratis memenuhi syarat | `standard`,`st1`,`sc1`,`gp2`, dan `gp3` | `standard`,`st1`,`sc1`,`gp2`, dan `gp3` |
| AMI bertanda Tingkat gratis memenuhi syarat | Periksa AMI yang bertanda Tingkat gratis memenuhi syarat. | Periksa AMI yang bertanda Tingkat gratis memenuhi syarat. |
| Batas penggunaan | Dibatasi pada batas penggunaan, setelah itu Anda akan ditagih tarif bayar sesuai penggunaan. | Dapatkan kredit pendaftaran USD $100 dan dapatkan kredit tambahan hingga $100. |
| Durasi Tingkat Gratis | Tingkat Gratis Anda berlaku selama 12 bulan sejak tanggal Anda membuat akun. Selama waktu ini, jika Anda melebihi batas penggunaan Tingkat Gratis, Anda akan ditagih tarif bayar sesuai penggunaan. | Tingkat Gratis Anda berlaku selama 6 bulan sejak tanggal Anda membuat akun, atau ketika kredit Anda habis, mana yang terjadi lebih dulu. Anda tidak dapat melebihi batas Tingkat Gratis Anda. |

Untuk membuat daftar jenis instans yang memenuhi syarat tingkat gratis

Gunakan perintah describe-instance-types dengan filter. `free-tier-eligible`

```
aws ec2 describe-instance-types \
    --filters Name=free-tier-eligible,Values=true \
    --query "InstanceTypes[*].[InstanceType]" \
    --output text | sort
```

Untuk membuat daftar AMI yang memenuhi syarat tingkat gratis

Gunakan perintah describe-images dengan filter. `free-tier-eligible`

```
aws ec2 describe-images \
    --filters Name=free-tier-eligible,Values=true \
    --query "Images[*].[ImageId]" \
    --output text | sort
```

## Lacak penggunaan Tingkat Gratis untuk akun yang dibuat sebelum 15 Juli 2025

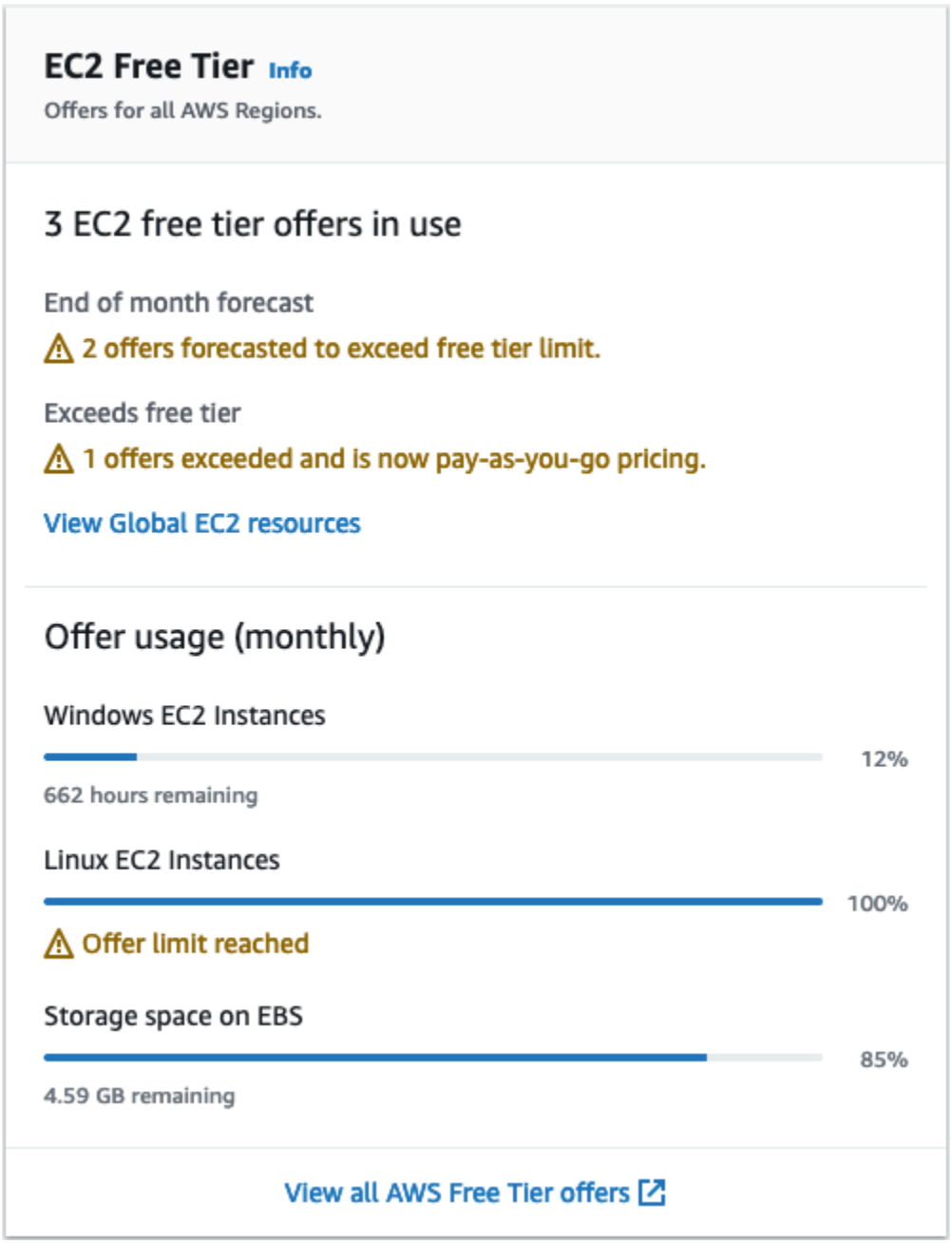
> **Note**
>
> Bagian ini hanya berlaku untuk pengguna Tingkat Gratis yang membuat Akun AWS sebelum 15 Juli 2025. Jika Anda membuat akun pada atau setelah 15 Juli 2025, lihat Mel acak penggunaan AWS Tingkat Gratis Anda di Panduan AWS Billing Pengguna.

Jika Anda membuat akun sebelum 15 Juli 2025, Anda dapat menggunakan Amazon EC2 tanpa dikenakan biaya jika Anda telah menjadi AWS pelanggan selama kurang dari 12 bulan dan Anda tetap dalam batas AWS Tingkat Gratis penggunaan. Penting untuk melacak penggunaan Tingkat Gratis Anda guna menghindari tagihan yang tidak terduga. Jika Anda melebihi batas Tingkat Gratis, Anda akan dikenai biaya bayar sesuai pemakaian standar. Untuk informasi selengkapnya, lihat AWS Tingkat Gratis.

> **Note**
>
> Jika Anda telah menjadi AWS pelanggan selama lebih dari 12 bulan, Anda tidak lagi memenuhi syarat untuk penggunaan Tingkat Gratis dan Anda tidak akan melihat kotak Tingkat Gratis EC2 yang dijelaskan dalam prosedur berikut.

Untuk melacak penggunaan Tingkat Gratis Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor EC2.
3. Temukan kotak Tingkat Gratis EC2 (di bagian kanan atas).




4. Di kotak Tingkat Gratis EC2, centang penggunaan Tingkat Gratis Anda, sebagai berikut:

   - Di bawah penawaran Tingkat Gratis EC2 yang digunakan, perhatikan peringatannya:
     - Prakiraan akhir bulan – Ini memberikan peringatan bahwa Anda akan dikenai biaya bulan ini jika melanjutkan dengan pola penggunaan saat ini.
     - Melebihi Tingkat Gratis – Ini memberikan peringatan bahwa Anda telah melebihi batas Tingkat Gratis dan Anda sudah dikenai biaya.

- Di bawah Penggunaan penawaran (bulanan), perhatikan penggunaan instans Linux, instans Windows, dan penyimpanan EBS Anda. Persentase menunjukkan jumlah batas Tingkat Gratis yang telah Anda gunakan bulan ini. Jika telah mencapai 100%, Anda akan dikenai biaya untuk penggunaan lebih lanjut.

> **Note**
>
> Informasi ini muncul hanya setelah Anda membuat instans. Namun, informasi penggunaan tidak diperbarui secara waktu nyata; informasi ini diperbarui tiga kali sehari.

5. Untuk menghindari biaya lebih lanjut, hapus sumber daya apa pun yang dikenai biaya saat ini, atau akan dikenai biaya jika Anda melebihi batas penggunaan Tingkat Gratis.

   - Untuk petunjuk untuk menghapus instans Anda, lihatMenghentikan instans Amazon EC2.
   - Untuk memeriksa apakah Anda memiliki sumber daya di Wilayah lain yang mungkin menimbulkan biaya, di kotak Tingkat Gratis EC2, pilih Lihat sumber daya EC2 Global untuk membuka Tampilan Global EC2. Untuk informasi selengkapnya, lihat Melihat sumber daya di seluruh Wilayah menggunakan AWS Tampilan Global.

6. Untuk melihat penggunaan sumber daya Anda untuk semua Layanan AWS di bawah AWS Tingkat Gratis, di bagian bawah kotak Tingkat Gratis EC2, pilih Lihat semua AWS Tingkat Gratis penawaran. Untuk informasi selengkapnya, lihat Mencoba layanan menggunakan AWS Tingkat Gratis di Panduan Pengguna Pen AWS agihan.

# Memahami kode untuk Amazon EC2 dalam laporan penagihan dan penggunaan

Saat Anda menggunakan Amazon EC2, kami menyertakan kode terkait dalam laporan AWS penagihan dan penggunaan Anda. Meninjau kode ini membantu Anda memahami biaya dan pola penggunaan untuk Amazon EC2. Melacak dan mengelola pengeluaran Anda sangat penting untuk mengoptimalkan biaya Anda.

Tabel berikut menjelaskan kode untuk Amazon EC2 yang muncul di laporan penagihan dan penggunaan Anda. Untuk daftar kode Wilayah yang digunakan dalam laporan penagihan dan penggunaan, lihat Kode penagihan AWS wilayah.

Kode penagihan untuk:

- Contoh
- Instans bare metal
- Host Khusus
- Instans Khusus
- Optimisasi EBS
- Reservasi Kapasitas

Sumber daya terkait

- the section called “Opsi penagihan dan pembelian”
- the section called “Memahami penagihan AMI”
- EC2 Harga Amazon

## Contoh

| Kode | Deskripsi | Unit |
| --- | --- | --- |
| *region*-BoxUsage :*instance-type* | Waktu berjalan untuk Instans Sesuai Permintaan. | Jam |
| *region*-HostBoxUsage :*instance-type* | Waktu berjalan untuk instans pada Host Khusus. | Jam |
| *region*-SpotUsage :*instance-type* | Waktu berjalan untuk Instans Spot. | Jam |

## Instans bare metal

| Kode | Deskripsi | Unit |
| --- | --- | --- |
| *region*-BoxUsage : *instance-family* .logam | Waktu berjalan untuk Instans On-Demand bare metal. | Jam |
| *region*-HostBoxUsage : *instance-family* .logam | Waktu berjalan untuk instans bare metal di Host Khusus. | Jam |
| *region*-SpotUsage : *instance-family* .logam | Waktu berjalan untuk Instans Spot bare metal. | Jam |

## Host Khusus

| Kode | Deskripsi | Unit |
| --- | --- | --- |
| *region*-HostUsage :*host-type* | Waktu dimana Dedicated Host disediakan. | Jam |
| *region*-ReservedHostUsage :*host-type* | Waktu Reservasi Tuan Rumah Khusus diterapkan. | Jam |

## Instans Khusus

| Kode | Deskripsi | Unit |
| --- | --- | --- |
| *region*-DedicatedUsage :*instance-type* | Waktu berjalan untuk Instans Khusus. | Jam+Biaya per wilayah |

## Optimisasi EBS

| Kode | Deskripsi | Unit |
| --- | --- | --- |
| *region*-EBSOptimized :*instance-type* | Waktu optimasi EBS diaktifkan. | Jam |

## Reservasi Kapasitas

| Kode | Deskripsi | Unit |
| --- | --- | --- |
| *region*-Reservation :*instance-type* | Waktu instans yang dicadangkan untuk Reservasi Kapasitas. | Jam |
| *region*-UnusedBox :*instance-type* | Waktu instans cadangan yang tidak digunakan untuk Reservasi Kapasitas. | Jam |
| *region*-DedicatedRes :*instance-type* | Waktu instans yang dipesan untuk Reservasi Kapasitas Khusus. | Jam |
| *region*-UnusedDed :*instance-type* | Waktu instans cadangan yang tidak digunakan untuk Reservasi Kapasitas Khusus. | Jam |

# Buat inventaris instans EC2 Anda

Amazon EC2 menyediakan kapasitas komputasi yang dapat diskalakan sesuai permintaan. Instance EC2 adalah server virtual yang berjalan di AWS jaringan.

Tabel berikut menjelaskan karakteristik utama dari instance EC2. Administrator sistem dapat menggunakan panduan yang diberikan dalam panduan ini untuk mendapatkan informasi tentang

karakteristik ini dan menggunakannya untuk mengonfigurasi server yang setara secara fungsional yang berjalan di lokasi atau di jaringan Penyedia Cloud lain.

| Karakteristik | Deskripsi |
| --- | --- |
| Lokasi geografis | Amazon EC2 dihosting di semua Wilayah di AWS seluruh dunia. Anda memilih lokasi untuk instans yang menempatkannya dekat dengan pelanggan Anda. Anda dapat meluncurkan instance di beberapa lokasi. |
| Spesifikasi perangkat keras | Amazon EC2 menyediakan berbagai pilihan jenis instans, dioptimalkan untuk berbagai kasus penggunaan. Jenis instans yang Anda pilih untuk instans menentukan sumber daya komputasi, memori, jaringan, dan penyimpan annya. |
| Gambar | Image Mesin Amazon (AMI) menyertakan perangkat lunak untuk diinstal pada instance Anda saat diluncurkan. Ini termasuk sistem operasi, paket perangkat lunak, dan konfigurasi khusus. |
| Alamat IP dan nama host DNS | Instans menerima alamat IP pribadi dan nama host DNS pribadi. Jika Anda mengonfigurasi alamat IP publik untuk sebuah instance, itu juga menerima nama host DNS publik. |
| Aturan grup keamanan | Aturan grup keamanan yang terkait dengan instans Anda menentukan lalu lintas masuk dan lalu lintas keluar mana yang diizinkan. |
| Data pengguna | Data pengguna tersedia untuk instans saat peluncuran. Ini terdiri dari skrip shell (Linux) atau PowerShell skrip (Windows). |

# Lokasi geografis

Amazon EC2 tersedia di setiap Wilayah di AWS seluruh dunia. Setiap Wilayah adalah wilayah geografis yang terpisah. Anda dapat menurunkan latensi jaringan saat memilih Wilayah untuk server yang dekat dengan sebagian besar pengguna Anda.

Anda dapat menggunakan Amazon EC2 Global View untuk membuat daftar instans Amazon EC2 di semua Wilayah. Dengan Konsol Manajemen AWS dan API, Anda harus mencantumkan instance untuk setiap Wilayah secara individual.

Mengapa hal ini penting

Setelah menentukan lokasi instans, Anda dapat memutuskan apakah akan menerapkan server yang setara secara fungsional di lokasi yang sama atau lokasi yang berbeda, tergantung pada kebutuhan Anda.

Untuk mendapatkan ringkasan instans EC2 Anda di semua Wilayah

1. Buka konsol Amazon EC2 Global View di https://console.aws.amazon.com/ec2globalview/home.
2. Pada tab Penjelajah wilayah, di bawah Ring kasan, periksa jumlah sumber daya untuk Instans, yang mencakup jumlah instance dan jumlah Wilayah. Pilih teks yang digarisbawahi untuk melihat bagaimana jumlah instans tersebar di seluruh Wilayah.
3. Pada tab Pencarian global, pilih filter klien Jenis sumber daya = Instance. Anda dapat memfilter hasil lebih lanjut dengan menentukan Region atau tag.

Untuk mendapatkan jumlah instans EC2 di Wilayah menggunakan AWS CLI

Gunakan perintah describe-instance berikut untuk menghitung instance di Wilayah yang ditentukan. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance.

```
aws ec2 describe-instances \
    --region us-east-2 \
    --query "length(Reservations[*].Instances[])"
```

Berikut ini adalah output contoh.

```
27
```

Untuk mendapatkan daftar ID instance alih-alih jumlah instance, gunakan `--query` parameter berikut sebagai gantinya.

```
--query "Reservations[*].Instances[].InstanceId"
```

## Spesifikasi perangkat keras

Jenis instans yang Anda tentukan untuk instans EC2 menentukan sumber daya komputasi, memori, penyimpanan, dan jaringan yang tersedia untuknya. Setiap rangkaian instans menawarkan keseimbangan sumber daya komputasi, memori, jaringan, dan penyimpanan yang berbeda. Untuk informasi selengkapnya, lihat Panduan Jenis Instans Amazon EC2.

Mengapa ini penting

Setelah menentukan spesifikasi perangkat keras untuk instans Anda, Anda dapat menentukan spesifikasi minimum yang diperlukan untuk server yang setara secara fungsional.

Untuk mendapatkan ringkasan jenis instans yang digunakan oleh instans Anda

Gunakan perintah describe-instance berikut. Per sed intah menghapus garis dengan tanda kurung dan baris kosong. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance.

```
aws ec2 describe-instances \
    --query "Reservations[*].Instances[].InstanceType" | sed 's/[][]//g;/^$/d' | sort |
 uniq -c | sort -nr
```

Berikut ini adalah output contoh.

```
  20    "c6i.4xlarge",
   5    "t2.micro",
   2    "g6e.2xlarge",
```

Untuk mendapatkan informasi tentang jenis instans tertentu

Gunakan perintah describe-instance-types berikut untuk menjelaskan jenis instance. `--query`Parameter hanya mencakup bidang yang ditentukan dalam output. Ini adalah karakteristik dasarnya. Anda dapat menyertakan bidang tambahan apa pun yang Anda butuhkan, atau Anda dapat merujuk ke spesifikasi jenis instans Amazon EC2 di Panduan Jenis Instans Amazon EC2.

```
aws ec2 describe-instance-types \
    --instance-types c6i.4xlarge t2.micro g6e.2xlarge \
    --query "InstanceTypes[*].
[InstanceType,VCpuInfo.DefaultVCpus,MemoryInfo.SizeInMiB,NetworkInfo.NetworkPerformance]"
 \
    --output table
```

Berikut ini adalah output contoh. Kolom adalah tipe instance, vCPU, memori (GiB), dan bandwidth jaringan (Gbps).

```
-----------------------------------------------------
|               DescribeInstanceTypes               |
+--------------+-----+--------+----------------------+
|  t2.micro    |  1  |  1024  |  Low to Moderate     |
|  c6i.4xlarge |  16 |  32768 |  Up to 12.5 Gigabit  |
|  g6e.2xlarge |  8  |  65536 |  Up to 20 Gigabit    |
+--------------+-----+--------+----------------------+
```

## Image

Amazon Machine Image (AMI) menyediakan perangkat lunak yang diinstal pada instans Anda saat diluncurkan, seperti sistem operasi, layanan, alat pengembangan, dan aplikasi. Anda dapat mengekspor AMI ke format lain untuk digunakan dengan server lain.

Mengapa ini penting

Setelah menentukan AMI untuk instans, Anda dapat merencanakan gambar yang diperlukan untuk server yang setara secara fungsional. Dimungkinkan untuk mengekspor AMI Anda ke format yang dapat Anda gunakan di tempat lain. Atau, Anda mungkin perlu menyambungkan ke instance yang Anda luncurkan dari setiap AMI, mengonfirmasi apa yang diinstal dan bagaimana konfigurasi, dan memastikan bahwa Anda memiliki salinan semua yang Anda butuhkan.

Untuk mendapatkan AMI untuk instans Anda

Gunakan perintah describe-instance berikut. Per sed intah menghapus garis dengan tanda kurung dan baris kosong. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance.

```
aws ec2 describe-instances \
```

```
    --query "Reservations[*].Instances[].ImageId" | sed 's/[][]//g;/^$/d' | sort | uniq
 -c | sort -nr
```

Berikut ini adalah output contoh.

```
20     "ami-0a70b9d193ae8a79",
 5     "ami-07d9cf938edb0739b",
 2     "ami-09245d5773578a1d6",
```

Untuk mendapatkan informasi tentang AMI tertentu

Gunakan perintah describe-images berikut. `--query`Parameter hanya mencakup bidang yang ditentukan dalam output. Anda dapat menyertakan bidang tambahan yang Anda butuhkan.

```
aws ec2 describe-images \
    --image-id ami-0a70b9d193ae8a79 ami-07d9cf938edb0739b ami-09245d5773578a1d6 \
    --query "Images[*].
{ID:ImageId,CreateDate:CreationDate,Name:Name,Description:Description}"
```

Berikut ini adalah output contoh.

```
[
    {
        "ID": "ami-0a70b9d193ae8a799",
        "CreateDate": "2025-03-01T02:22:41.000Z",
        "Name": "web-server-3.1",
        "Description": "Image for web servers"
    },
    {
        "ID": "ami-07d9cf938edb0739b",
        "CreateDate": "2025-02-01T23:59:03.000Z",
        "Name": "awesome-application-11.5",
        "Description": "Image for Awesome Application"
    },
    {
        "ID": "ami-09245d5773578a1d6",
        "CreateDate": "2025-01-31T02:22:41.000Z",
        "Name": "monitoring-4.2",
        "Description": "Monitoring software"
    }
]
```

Untuk membuat file VM dari AMI

Gunakan perintah export-image. Format gambar yang didukung adalah VHD (kompatibel dengan Citrix Xen dan Microsoft Hyper-V), VMDK (kompatibel dengan VMware ESX dan VMware vSphere), dan format mentah (kompatibel dengan hypervisor KVM dan Xen). Untuk informasi tentang persyaratan dan batasan VM Import/Export, lihat Import/Export persyaratan VM.

## Alamat IP dan nama host DNS

Pengguna Anda terhubung ke instans EC2 Anda melalui internet menggunakan nama host DNS publik mereka. Nama host DNS publik dari instance EC2 diselesaikan ke alamat IP publiknya. Instans EC2 dapat berkomunikasi satu sama lain menggunakan alamat IP pribadi mereka.

Mengapa ini penting

Alamat IP memungkinkan perangkat untuk berkomunikasi satu sama lain melalui jaringan lokal atau internet. Dengan mencatat alamat IP dan nama DNS instans Anda, Anda dapat membantu memastikan bahwa server yang setara secara fungsional dapat berkomunikasi dengan klien atau perangkat yang sama dengan instans Anda. Misalnya, Anda dapat memperbarui konfigurasi penyeimbang beban atau catatan DNS yang Anda buat untuk penyedia DNS Anda.

Untuk mendapatkan alamat IP dan nama host DNS untuk instans Anda

Gunakan perintah describe-instance berikut. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance. `--query`Parameter hanya mencakup bidang yang ditentukan dalam output. Anda dapat menyertakan bidang tambahan yang Anda butuhkan.

```
aws ec2 describe-instances \
    --query "Reservations[*].Instances[].[InstanceId,PrivateIpAddress,PublicDnsName]" \
    --output table
```

Berikut ini adalah output contoh. Kolom adalah ID instance, alamat IPv4 pribadi, dan nama DNS IPv4 publik.

```
----------------------------------------------------------------------------------
|                                DescribeInstances                               |
 |
+---------------------+-----------+-------------------------------------------------
+
| i-0bac57d7472c89bac | 10.0.2.139 | ec2-192-0-2-142.us-east-2.compute.amazonaws.com
 |
```

```
| i-0fa8b7678975a3fff | 10.0.14.85 | ec2-198-51-100-56.us-east-2.compute.amazonaws.com
 |
  ...
  ...
| i-08fd74f3f1595fdbd | 10.0.1.241 | ec2-203.0.113.13.us-east-2.compute.amazonaws.com
 |
+---------------------+------------+-------------------------------------------------
+
```

## Aturan-aturan grup keamanan

Grup keamanan bertindak sebagai firewall virtual. Setelah Anda mengaitkan grup keamanan dengan instans EC2, aturannya mengizinkan lalu lintas masuk dan keluar untuk instans EC2 melalui port dan protokol tertentu.

Mengapa ini penting

Setelah Anda menentukan lalu lintas masuk yang diizinkan untuk mencapai server Anda dan lalu lintas keluar diizinkan meninggalkan server Anda, Anda dapat merencanakan aturan firewall yang diperlukan untuk server yang setara secara fungsional.

Untuk mendapatkan grup keamanan untuk instans Anda

Gunakan perintah describe-instance berikut. Per sed intah menghapus garis dengan tanda kurung dan baris kosong. Anda harus menjalankan perintah ini di setiap Wilayah di mana Anda memiliki instance.

```
aws ec2 describe-instances \
    --query "Reservations[*].Instances[].SecurityGroups[].GroupId" | sed 's/[][]//g;/^
$/d' | sort | uniq -c | sort -nr
```

Berikut ini adalah output contoh.

```
27     "sg-01dd3383691d02f42",
10     "sg-08c77cc910c0b3b2c",
 2     "sg-00f4e409629f1a42d",
```

Untuk mendapatkan aturan masuk untuk grup keamanan

Gunakan perintah describe-security-group-rules berikut untuk menampilkan aturan untuk grup keamanan yang ditentukan di mana berada. `IsEgress` `false`

```
aws ec2 describe-security-group-rules \
    --filters Name=group-id,Values=sg-01dd3383691d02f42 \
    --query 'SecurityGroupRules[?IsEgress==`false`]'
```

Berikut ini adalah output contoh. Aturan pertama adalah aturan inbound default, yang memungkinkan lalu lintas masuk dari semua sumber daya yang ditetapkan ke grup keamanan ini. Aturan lainnya memungkinkan lalu lintas SSH dari alamat IP dalam daftar awalan yang ditentukan. Untuk menjelaskan blok CIDR dalam daftar awalan, gunakan perintah describe- prefix-lists.

```
[
    {
        "SecurityGroupRuleId": "sgr-06c8b42574a91db1b",
        "GroupId": "sg-01dd3383691d02f42",
        "GroupOwnerId": "123456789012",
        "IsEgress": false,
        "IpProtocol": "-1",
        "FromPort": -1,
        "ToPort": -1,
        "ReferencedGroupInfo": {
            "GroupId": "sg-01dd3383691d02f42",
            "UserId": "123456789012"
        },
        "Tags": [],
        "SecurityGroupRuleArn": "arn:aws:ec2:us-west-2:123456789012:security-group-
rule/sgr-06c8b42574a91db1b"
    },
    {
        "SecurityGroupRuleId": "sgr-0886a5d46afcd1758",
        "GroupId": "sg-01dd3383691d02f42",
        "GroupOwnerId": "123456789012",
        "IsEgress": false,
        "IpProtocol": "tcp",
        "FromPort": 22,
        "ToPort": 22,
        "PrefixListId": "pl-f8a6439125e7bf465",
        "Tags": [],
        "SecurityGroupRuleArn": "arn:aws:ec2:us-east-2:123456789012:security-group-
rule/sgr-0886a5d46afcd1758"
    }
]
```

Untuk mendapatkan aturan keluar untuk grup keamanan

Gunakan perintah describe-security-group-rules berikut untuk menampilkan aturan untuk grup keamanan yang ditentukan di mana berada. `IsEgress` `true`

```
aws ec2 describe-security-group-rules \
    --filters Name=group-id,Values=sg-01dd3383691d02f42 \
    --query 'SecurityGroupRules[?IsEgress==`true`]'
```

Berikut ini adalah output contoh. Ini hanya berisi aturan keluar default, yang memungkinkan semua lalu lintas IPv4 keluar.

```
[
    {
        "SecurityGroupRuleId": "sgr-048f09a719247dce7",
        "GroupId": "sg-01dd3383691d02f42",
        "GroupOwnerId": "123456789012",
        "IsEgress": true,
        "IpProtocol": "-1",
        "FromPort": -1,
        "ToPort": -1,
        "CidrIpv4": "0.0.0.0/0",
        "Tags": [],
        "SecurityGroupRuleArn": "arn:aws:ec2:us-east-2:123456789012:security-group-
rule/sgr-048f09a719247dce7"
    }
]
```

## Data pengguna

Saat meluncurkan instance EC2, Anda dapat meneruskan skrip shell ke instance menggunakan data pengguna. Perhatikan bahwa data pengguna dikodekan base64, jadi Anda perlu memecahkan kode data pengguna untuk membaca skrip.

Mengapa hal ini penting

Jika menjalankan perintah saat peluncuran sebagai bagian dari pengaturan instans, Anda mungkin perlu melakukan tugas yang sama saat menyiapkan server yang setara secara fungsional.

Untuk melihat data pengguna yang didekodekan untuk sebuah instance

Gunakan perintah describe-inst ance-atribut berikut. Per base64 intah memecahkan kode data pengguna.

```
aws ec2 describe-instance-attribute \
    --instance-id i-1234567890abcdef0 \
    --attribute userData \
    --output text \
    --query "UserData.Value" | base64 --decode
```

Berikut ini adalah output contoh.

```
#!/bin/bash
yum update -y
service httpd start
chkconfig httpd on
```

## Sumber daya terkait

Berikut ini adalah karakteristik tambahan dari instans EC2:

- Pasangan kunci
- Penyimpanan
- Tanda

Anda dapat memverifikasi apakah Anda menggunakan yang berikut ini untuk meluncurkan instans EC2 atau mendistribusikan lalu lintas antara instans EC2 Anda:

- Grup penskalaan otomatis
- Armada EC2
- Elastic Load Balancing

# Memecahkan masalah dengan instans Amazon EC2

Prosedur dan tip berikut dapat membantu Anda memecahkan masalah dengan instans Amazon EC2 Anda.

Masalah

- Memecahkan masalah peluncuran instans Amazon EC2
- Memecahkan masalah penghentian instans Amazon EC2
- Memecahkan masalah penghentian instans Amazon EC2
- Memecahkan masalah instans Amazon EC2 yang tidak dapat dijangkau
- Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux
- Memecahkan masalah instans Amazon EC2 Linux dengan pemeriksaan status yang gagal
- Memecahkan masalah booting instans Amazon EC2 Linux dari volume yang salah
- Memecahkan masalah saat menghubungkan ke instans Windows Amazon EC2 Anda
- Memecahkan masalah mulai instans Amazon EC2 Windows
- Memecahkan masalah dengan instans Amazon EC2 Windows
- Kernel debugging untuk instance Windows melalui jaringan
- Setel ulang kata sandi administrator Windows untuk instans Windows Amazon EC2
- Memecahkan masalah Sysprep dengan instans Amazon EC2 Windows
- Memecahkan masalah instans Amazon EC2 Linux yang rusak menggunakan EC2Rescue
- Memecahkan masalah instans Windows Amazon EC2 yang rusak menggunakan EC2Rescue
- EC2 Serial Console untuk instans
- Mengirim interupsi diagnostik untuk men-debug instans Amazon EC2 yang tidak dapat dijangkau

## Memecahkan masalah peluncuran instans Amazon EC2

Berikut ini adalah kiat pemecahan masalah untuk membantu Anda memecahkan masalah saat meluncurkan instans Amazon EC2.

Masalah Peluncuran

- Nama perangkat tidak valid

- Batas instans terlampaui
- Kapasitas instans tidak cukup
- Konfigurasi yang diminta saat ini tidak didukung. Periksa dokumentasi untuk konfigurasi yang didukung.
- Instans langsung terhenti
- Izin tidak cukup
- Penggunaan CPU tinggi segera setelah Windows dimulai (hanya instans Windows)
- Meluncurkan IMDSv1-enabled instance gagal

## Nama perangkat tidak valid

### Deskripsi

Anda mendapatkan kesalahan `Invalid device name` *`device_name`* saat mencoba meluncurkan instans baru.

### Penyebab

Jika Anda mendapatkan kesalahan ini saat mencoba meluncurkan instans, nama perangkat yang ditentukan untuk satu atau beberapa volume dalam permintaan memiliki nama perangkat yang tidak valid. Kemungkinan penyebabnya meliputi:

- Nama perangkat mungkin digunakan oleh AMI yang dipilih.
- Nama perangkat mungkin dipesan untuk volume root.
- Nama perangkat mungkin digunakan untuk volume lain dalam permintaan.
- Nama perangkat mungkin tidak valid untuk sistem operasi.

### Solusi

Untuk mengatasi masalah ini:

- Pastikan nama perangkat tidak digunakan di AMI yang Anda pilih. Jalankan perintah berikut untuk menampilkan nama perangkat yang digunakan oleh AMI.

```
aws ec2 describe-images --image-id ami-0abcdef1234567890 --query
 'Images[*].BlockDeviceMappings[].DeviceName'
```

- Pastikan Anda tidak menggunakan nama perangkat yang dipesan untuk volume root. Untuk informasi selengkapnya, lihat Nama perangkat yang tersedia.
- Pastikan setiap volume yang ditentukan dalam permintaan Anda memiliki nama perangkat yang unik.
- Pastikan nama perangkat yang Anda tentukan berada menggunakan format yang benar. Untuk informasi selengkapnya, lihat Nama perangkat yang tersedia.

## Batas instans terlampaui

### Deskripsi

Anda mendapatkan kesalahan `InstanceLimitExceeded` saat mencoba meluncurkan instans baru atau memulai ulang instans yang terhenti.

### Penyebab

Jika Anda mendapatkan kesalahan `InstanceLimitExceeded` saat mencoba meluncurkan instans baru atau memulai ulang instans yang terhenti, Anda telah mencapai batas jumlah instans yang dapat Anda luncurkan di Wilayah. Saat Anda membuat AWS akun, kami menetapkan batas default pada jumlah instans yang dapat Anda jalankan berdasarkan Per-wilayah.

### Solusi

Anda dapat meminta kenaikan batas instans berdasarkan wilayah. Untuk informasi selengkapnya, lihat Kuota layanan Amazon EC2.

## Kapasitas instans tidak cukup

### Deskripsi

Anda mendapatkan kesalahan `InsufficientInstanceCapacity` saat mencoba meluncurkan instans baru atau memulai ulang instans yang terhenti.

### Penyebab

Jika Anda mendapatkan kesalahan ini saat mencoba meluncurkan instans atau memulai ulang instans yang dihentikan, saat ini AWS tidak memiliki On-Demand kapasitas yang cukup untuk memenuhi permintaan Anda.

### Solusi

Untuk mengatasi masalah ini, coba lakukan hal berikut:

- Tunggu beberapa menit, lalu kirim permintaan Anda lagi; kapasitas sering kali dapat berubah.
- Kirim permintaan baru dengan jumlah instans yang lebih sedikit. Misalnya, jika Anda membuat permintaan tunggal untuk meluncurkan 15 instans, cobalah membuat 3 permintaan untuk 5 instans, atau 15 permintaan untuk 1 instans.
- Jika Anda meluncurkan instans, kirimkan permintaan baru tanpa menentukan Zona Ketersediaan.
- Jika Anda meluncurkan instans, kirimkan permintaan baru menggunakan tipe instans yang berbeda (yang dapat diubah ukurannya di tahap berikutnya). Untuk informasi selengkapnya, lihat Perubahan jenis instans Amazon EC2.
- Jika Anda meluncurkan instans ke grup penempatan klaster, Anda bisa mendapatkan kesalahan kapasitas yang tidak memadai.

## Konfigurasi yang diminta saat ini tidak didukung. Periksa dokumentasi untuk konfigurasi yang didukung.

### Deskripsi

Anda mendapatkan kesalahan `Unsupported` saat mencoba meluncurkan instans baru karena konfigurasi instans tidak didukung.

### Penyebab

Pesan kesalahan memberikan detail tambahan. Misalnya, tipe instans atau opsi pembelian instans mungkin tidak didukung di dalam Wilayah atau Zona Ketersediaan tertentu.

### Solusi

Coba konfigurasi instans yang berbeda. Untuk mencari tipe instans yang memenuhi persyaratan Anda, lihat Menemukan tipe instans Amazon EC2.

## Instans langsung terhenti

### Deskripsi

Instans Anda berubah dari status `pending` menjadi status `terminated`.

## Penyebab

Berikut ini adalah beberapa alasan instans dapat langsung terhenti:

- Anda telah melebihi batas volume EBS. Untuk informasi selengkapnya, lihat Batas volume Amazon EBS untuk instans Amazon EC2.
- Snapshot EBS rusak.
- Volume EBS root terenkripsi dan Anda tidak memiliki izin guna mengakses kunci KMS untuk dekripsi.
- Snapshot yang ditentukan dalam pemetaan perangkat blok untuk AMI dienkripsi dan Anda tidak memiliki izin guna mengakses kunci KMS untuk dekripsi atau Anda tidak memiliki akses ke kunci KMS untuk mengenkripsi volume yang dipulihkan.
- Amazon S3-backed AMI yang Anda gunakan untuk meluncurkan instance tidak memiliki bagian yang diperlukan (file image.part. xx  berkas).

Untuk informasi selengkapnya, dapatkan alasan penghentian menggunakan salah satu metode berikut.

Untuk mendapatkan alasan penghentian menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, dan pilih instans.
3. Di tab pertama, cari alasannya di samping Alasan transisi status.

Untuk mendapatkan alasan penghentian menggunakan AWS CLI

1. Gunakan perintah describe-instances dan tentukan ID instans.

```
aws ec2 describe-instances --instance-id i-1234567890abcdef0
```

2. Tinjau respons JSON yang dikembalikan oleh perintah dan perhatikan nilainya di elemen respons `StateReason`.

   Blok kode berikut ini menunjukkan contoh elemen respons `StateReason`.

```
"StateReason": {
  "Message": "Client.VolumeLimitExceeded: Volume limit exceeded",
  "Code": "Server.InternalError"
```

```
},
```

Untuk mendapatkan alasan penghentian menggunakan AWS CloudTrail

Untuk informasi selengkapnya, lihat Melihat CloudTrail acara dengan riwayat acara di Panduan AWS CloudTrail Pengguna.

### Solusi

Dengan bergantung pada alasan penghentian, lakukan salah satu tindakan berikut:

- **`Client.VolumeLimitExceeded: Volume limit exceeded`** – Hapus volume yang tak terpakai. Anda dapat mengirim permintaan untuk meningkatkan batas volume.
- **`Client.InternalError: Client error on launch`**— Pastikan bahwa Anda memiliki izin yang diperlukan untuk mengakses yang AWS KMS keys digunakan untuk mendekripsi dan mengenkripsi volume. Untuk informasi selengkapnya, lihat Menggunakan kebijakan kunci di AWS KMS dalam Panduan Developer AWS Key Management Service .

## Izin tidak cukup

### Deskripsi

Anda mendapatkan kesalahan `"errorMessage": "You are not authorized to perform this operation."` saat mencoba meluncurkan instans baru, dan peluncuran tersebut gagal.

### Penyebab

Jika Anda mendapatkan kesalahan ini saat mencoba meluncurkan instans, Anda tidak memiliki izin IAM yang diperlukan untuk meluncurkan instans tersebut.

Kemungkinan izin yang ada mencakup:

- `ec2:RunInstances`
- `iam:PassRole`

Izin lain mungkin juga tidak ada. Untuk daftar izin yang diperlukan guna meluncurkan instans, lihat contoh kebijakan IAM di bawah Contoh: Menggunakan wizard peluncuran instans EC2 dan Luncurkan instance (RunInstances).

### Solusi

Untuk mengatasi masalah ini:

- Jika Anda membuat permintaan sebagai pengguna IAM, verifikasi bahwa Anda memiliki izin berikut:
  - `ec2:RunInstances` dengan sumber daya wildcard ("*")
  - `iam:PassRole` dengan sumber daya yang cocok dengan ARN peran (misalnya, `arn:aws:iam::999999999999:role/ExampleRoleName`)
- Jika Anda tidak memiliki izin sebelumnya, edit kebijakan IAM yang terkait dengan peran atau pengguna IAM untuk menambahkan izin penting yang tidak ada.

Jika masalah tidak teratasi dan Anda terus menerima kesalahan kegagalan peluncuran, Anda dapat melakukan dekode pesan kegagalan otorisasi yang disertakan dalam kesalahan. Pesan yang didekodekan mencakup izin yang tidak ada dari kebijakan IAM. Untuk informasi selengkapnya, lihat Bagaimana cara memecahkan kode pesan kegagalan otorisasi setelah saya menerima kesalahan "UnauthorizedOperation" selama peluncuran instans EC2?

## Penggunaan CPU tinggi segera setelah Windows dimulai (hanya instans Windows)

> **Note**
>
> Tip pemecahan masalah ini hanya untuk instans Windows.

Jika Pembaruan Windows diatur ke Periksa pembaruan, tetapi biarkan saya memilih apakah akan mengunduh dan menginstalnya (pengaturan instans default), pemeriksaan ini dapat menghabiskan sekitar 50 - 99% CPU pada instans. Jika penggunaan CPU ini menyebabkan masalah pada aplikasi, Anda dapat mengubah pengaturan Pembaruan Windows secara manual di Panel Kontrol atau Anda dapat menggunakan skrip berikut di bidang data pengguna Amazon EC2:

```
reg add "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\WindowsUpdate\Auto Update" /v
 AUOptions /t REG_DWORD /d 3 /f net stop wuauserv net start wuauserv
```

Saat Anda menjalankan skrip ini, tentukan nilai untuk /d. Nilai default-nya adalah 3. Kemungkinan nilainya mencakup berikut ini:

1. Jangan pernah memeriksa pembaruan
2. Periksa pembaruan, tetapi biarkan saya memilih apakah akan mengunduh dan menginstalnya
3. Unduh pembaruan, tetapi biarkan saya memilih apakah akan menginstalnya
4. Instal pembaruan secara otomatis

Setelah Anda memodifikasi data pengguna untuk instans, Anda dapat menjalankannya. Untuk informasi selengkapnya, lihat Menjalankan perintah pada instance Windows Anda saat peluncuran.

## Meluncurkan IMDSv1-enabled instance gagal

### Deskripsi

Anda mendapatkan peng `UnsupportedOperation` ecualian dengan pesan berikut:

```
You can't launch instances with IMDSv1 because httpTokensEnforced
is enabled for this account. Either launch the instance with
httpTokens=required or contact your account owner to disable
httpTokensEnforced using the ModifyInstanceMetadataDefaults API or the
account settings in the EC2 console.
```

### Penyebab

Kesalahan ini muncul saat Anda mencoba meluncurkan instans baru agar IMDSv1 diaktifkan (`httpTokens = optional`) di akun di mana pengaturan akun EC2 atau kebijakan deklaratif AWS Organisasi memberlakukan penggunaan IMDSv2 (). `httpTokensEnforced = enabled`

### Solusi

Jika Anda siap menggunakan IMDSv2 saja, luncurkan instans Anda dengan IMDSv1 disabled (). `httpTokens = required` Untuk memeriksa apakah Anda siap, lihatTransisi ke penggunaan Layanan Metadata Instans Versi 2.

Jika Anda masih memerlukan dukungan IMDSv1 pada instans baru atau yang sudah ada, Anda harus menonaktifkan penegakan IMDSv2 untuk akun di Wilayah. Untuk menonaktifkan penegakan IMDSv2, setel ke. `HttpTokensEnforced disabled` Untuk informasi selengkapnya, lihat ModifyInstanceMetadataDefaults Referensi API Amazon EC2. Jika Anda lebih suka mengkonfigurasi pengaturan ini menggunakan konsol, lihatMenerapkan IMDSv2 di tingkat akun.

# Memecahkan masalah penghentian instans Amazon EC2

Jika EBS-backed instans Amazon Anda tampak macet dalam `stopping` status, masalahnya mungkin ada pada komputer host yang mendasarinya.

Untuk mengatasi masalah ini, ikuti langkah-langkah berikut:

1. Hentikan instans secara paksa

   Gunakan konsol Amazon EC2 atau AWS CLI untuk menghentikan instans secara paksa. Untuk langkah-langkahnya, lihatMemaksa menghentikan instance.

   Instance pertama-tama akan mencoba shutdown yang anggun, yang mencakup pembilasan cache sistem file dan metadata (meskipun Anda secara opsional dapat melewati shutdown yang anggun). Jika shutdown yang anggun gagal diselesaikan dalam periode batas waktu, instance dimatikan secara paksa tanpa menghapus cache dan metadata sistem file.

2. Setelah paksa berhenti

   Lakukan prosedur pemeriksaan dan perbaikan sistem file.

   > **Important**
   >
   > Melakukan prosedur ini sangat penting karena penghentian paksa mencegah pembilasan cache sistem file dan metadata.

3. Jika paksa berhenti gagal

   Jika, setelah 10 menit, instance belum berhenti, lakukan hal berikut:

   a. Posting permintaan bantuan di AWS re:Post. Untuk membantu mempercepat resolusi, sertakan ID instans, dan jelaskan langkah-langkah yang telah Anda ambil.

   b. Atau, jika Anda memiliki paket dukungan, buat kasus dukungan teknis dalam Pusat Dukungan.

   c. Sambil menunggu bantuan, Anda dapat membuat instance pengganti jika diperlukan. Untuk langkah-langkahnya, lihat(Opsional) Buat instance pengganti.

Tidak ada biaya untuk penggunaan instans selagi instans dalam status `stopping` atau dalam status lain kecuali, `running`. Anda tidak dikenai biaya untuk penggunaan instans saat instans dalam status `running`.

Daftar Isi


- Memaksa menghentikan instance
- (Opsional) Buat instance pengganti


## Memaksa menghentikan instance

Anda dapat memaksa instance untuk berhenti. Jika setelah 10 menit instans tidak berhenti, unggah permintaan bantuan AWS re:Post. Untuk membantu mempercepat resolusi, sertakan ID instans, dan jelaskan langkah-langkah yang telah Anda ambil. Atau, jika Anda memiliki paket dukungan, buat kasus dukungan teknis dalam Pusat Dukungan.

> **Note**
>
> Dengan menggunakan konsol, Anda dapat memaksa instance untuk berhenti sementara instance dalam `stopping` status saja. Dengan menggunakan AWS CLI, Anda dapat memaksa instance berhenti saat instance berada di status`pending`,`running`, atau`stopping`.

Console

Untuk menghentikan instans secara paksa

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, kemudian pilih instans yang macet.
3. Pilih Status Inst ance, Force stop instance.

   Perhatikan bahwa Penghentian paksa instans hanya tersedia di konsol jika instans Anda dalam status `stopping`. Jika instance Anda berada dalam status lain (kecuali `shutting-down` dan`terminated`), Anda dapat menggunakan AWS CLI untuk menghentikan instans Anda secara paksa.
4. (Opsional) Untuk melewati shutdown OS yang anggun selama penghentian paksa, pilih kotak centang Le wati penutupan OS.

5. Pilih Force stop.

AWS CLI

Untuk menghentikan instans secara paksa

Gunakan perintah stop-instance dengan opsi. `--force`

```
aws ec2 stop-instances \
    --instance-ids i-1234567890abcdef0 \
    --force
```

Untuk melewati shutdown OS yang anggun selama force stop, sertakan `--skip-os-shutdown` opsi.

```
aws ec2 stop-instances \
    --instance-ids i-1234567890abcdef0 \
    --force \
    --skip-os-shutdown
```

PowerShell

Untuk menghentikan instans secara paksa

Gunakan Stop-EC2Instance cmdlet dan set `-Enforce` el ke. `true`

```
Stop-EC2Instance `
    -InstanceId i-1234567890abcdef0 `
    -Enforce $true
```

Untuk melewati shutdown OS yang anggun selama force stop, sertakan`-SkipOsShutdown` `$true`.

```
Stop-EC2Instance `
    -InstanceId i-1234567890abcdef0 `
    -Enforce $true `
    -SkipOsShutdown $true
```

# (Opsional) Buat instance pengganti

Saat menunggu bantuan dari AWS re:Post atau Pusat Dukungan, Anda dapat membuat instance pengganti jika diperlukan. Buat AMI dari instance macet, dan luncurkan instance baru menggunakan AMI baru.

> **Important**
>
> Anda dapat membuat instance pengganti jika instans macet hanya menghasilkan pemeriksaan status sistem, karena pemeriksaan status instance akan mengakibatkan AMI menyalin replika yang tepat dari sistem operasi yang rusak. Setelah Anda mengonfirmasi pesan status, buat AMI dan luncurkan instance baru menggunakan AMI baru.

Console

Untuk membuat instance pengganti

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, kemudian pilih instans yang macet.
3. Pilih Tindakan, Gambar dan templat, Buat gambar.
4. Pada halaman Buat gambar, lakukan hal berikut:
   a. Masukkan nama dan deskripsi untuk AMI.
   b. Hapus contoh Reboot.
   c. Pilih Buat gambar.

   Untuk informasi selengkapnya, lihat the section called “Membuat AMI dari instance”.
5. Luncurkan instans baru dari AMI dan verifikasi bahwa instans baru tersebut berfungsi.
6. Pilih instance macet, dan pilih Ac tions, Instance state, Terminate (delete) instance. Jika instans tersebut juga mengalami kemacetan saat pengakhiran, Amazon EC2 akan mengakhiri paksa secara otomatis dalam beberapa jam.

Jika Anda tidak dapat membuat AMI dari instans seperti yang dijelaskan dalam prosedur sebelumnya, Anda dapat mengatur instans pengganti sebagai berikut:

(Alternatif) Untuk membuat instans pengganti menggunakan konsol

1. Pilih instans dan pilih Deskripsi, Perangkat blok. Pilih setiap volume dan catat ID volumenya. Pastikan untuk mencatat volume yang merupakan volume root.
2. Di panel navigasi, pilih Volume. Pilih setiap volume untuk instans tersebut, lalu pilih Tindakan, Buat Snapshot.
3. Di panel navigasi, pilih Snapshot. Pilih snapshot yang baru saja Anda buat, lalu pilih Tindakan, Buat Volume.
4. Luncurkan instans menggunakan sistem operasi yang sama dengan instans yang macet. Catat ID volume dan nama perangkat volume root-nya.
5. Di panel navigasi, pilih Instans, pilih instans yang baru saja Anda luncurkan, lalu pilih Status instans, Hentikan instans.
6. Di panel navigasi, pilih Volume, pilih volume root dari instans yang dihentikan, lalu pilih Tindakan, Copot Volume.
7. Pilih volume root yang Anda buat dari instans yang macet, pilih Tindakan, Lampirkan Volume, dan lampirkan ke instans yang baru sebagai volume root-nya (menggunakan nama perangkat yang Anda catat). Lampirkan volume non-root tambahan ke instans.
8. Di panel navigasi, pilih Instans, lalu pilih instans pengganti. Pilih Status instans, Mulai instans. Verifikasi bahwa instans berfungsi.
9. Pilih instance macet, pilih Status  Instance,  Hentikan (hapus) instance. Jika instans tersebut juga mengalami kemacetan saat pengakhiran, Amazon EC2 akan mengakhiri paksa secara otomatis dalam beberapa jam.

AWS CLI

Untuk membuat instance pengganti

1. Buat AMI dari instance macet menggunakan perintah create-image dengan opsi. `--no-reboot`

```
aws ec2 create-image \
    --instance-id i-1234567890abcdef0 \
    --name "my-replacement-ami" \
    --description "AMI for replacement instance" \
    --no-reboot
```

2. Luncurkan instance baru dari AMI yang baru saja Anda buat, menggunakan perintah run-instance.
3. Verifikasi bahwa instans baru berfungsi.
4. (Opsional) Hentikan instance macet menggunakan perintah terminate-instance.

```
aws ec2 terminate-instances --instance-ids i-1234567890abcdef0
```

PowerShell

Untuk membuat instance pengganti

1. Buat AMI dari instance macet menggunakan New-EC2Image cmdlet dan set `-NoReboot` el ke. `true`

```
New-EC2Image `
    -InstanceId i-1234567890abcdef0 `
    -Name "my-replacement-ami" `
    -Description "AMI for replacement instance" `
    -NoReboot $true
```

2. Luncurkan instance baru dari AMI yang baru saja Anda buat, menggunakan New-EC2Instance cmdlet.
3. Verifikasi bahwa instans baru berfungsi.
4. (Opsional) Hentikan instance macet menggunakan Remove-EC2Instance cmdlet.

```
Remove-EC2Instance -InstanceId i-1234567890abcdef0
```

# Memecahkan masalah penghentian instans Amazon EC2

Mematikan atau menghapus instans Anda dikenal sebagai penghentian instans. Informasi berikut dapat membantu Anda memecahkan masalah saat Anda menghentikan instans.

Anda tidak ditagih atas penggunaan instans apa pun saat instans tidak berada dalam status `running`. Dengan kata lain, saat Anda menghentikan sebuah instans, Anda tidak lagi dibebani biaya untuk instans tersebut segera setelah statusnya berubah menjadi `shutting-down`.

## Instans langsung terhenti

Beberapa masalah dapat menyebabkan instans langsung terhenti pada saat memulai. Lihat Instans langsung terhenti untuk informasi selengkapnya.

## Penghentian instans yang tertunda

Jika instance Anda tetap dalam `shutting-down` status lebih dari beberapa menit, itu mungkin karena:

- Instance menjalankan skrip shutdown.
- Ada masalah dengan komputer host yang mendasarinya.

Setelah beberapa jam di `shutting-down` negara bagian, Amazon EC2 memperlakukan instance sebagai macet dan secara paksa menghentikannya.

Untuk menyelesaikan instans macet sendiri:

1. Memaksa menghentikan instance

   Gunakan konsol Amazon EC2 atau AWS CLI untuk menghentikan instans secara paksa. Untuk langkah-langkahnya, lihatMemaksa mengakhiri instance.

   Instance pertama-tama akan mencoba shutdown yang anggun, yang mencakup pembilasan cache sistem file dan metadata (meskipun Anda secara opsional dapat melewati shutdown yang anggun). Jika shutdown yang anggun gagal diselesaikan dalam periode batas waktu, instance dimatikan secara paksa tanpa menghapus cache dan metadata sistem file.
2. Jika paksa berhenti gagal

   Jika, setelah beberapa jam, instance belum dihentikan dan tampaknya macet berakhir, lakukan hal berikut:

   a. Posting permintaan bantuan di AWS re:Post. Untuk membantu mempercepat resolusi, sertakan ID instans, dan jelaskan langkah-langkah yang telah Anda ambil.

   b. Atau, jika Anda memiliki paket dukungan, buat kasus dukungan teknis dalam Pusat Dukungan.

## Memaksa mengakhiri instance

Jika instans Anda tampak macet saat dihentikan, Anda dapat memaksa instans untuk dihentikan. Jika, setelah beberapa jam, instance belum dihentikan, kirim permintaan bantuan ke AWS Re: post. Untuk membantu mempercepat resolusi, sertakan ID instans dan jelaskan langkah-langkah yang telah Anda ambil. Atau, jika Anda memiliki paket dukungan, buat kasus dukungan teknis dalam Pusat Dukungan.

### Console

Untuk memaksakan penghentian sebuah instance

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, kemudian pilih instans yang macet.
3. Pilih Status instans, Mem aksa mengakhiri instance.

   Perhatikan bahwa Force terminate instance hanya tersedia di konsol jika instans Anda berada dalam `stopping` status tersebut. Jika instance Anda berada dalam status lain (kecuali `shutting-down` dan`terminated`), Anda dapat menggunakan AWS CLI untuk memaksa mengakhiri instance Anda.
4. (Opsional) Untuk melewati shutdown OS yang anggun selama penghentian paksa, pilih kotak centang Le wati penutupan OS.
5. Pilih Paksa akhiri.

### AWS CLI

Untuk memaksakan penghentian sebuah instance

Gunakan perintah terminate-instance dengan opsi. `--force`

```
aws ec2 terminate-instances \
    --instance-ids i-1234567890abcdef0 \
    --force
```

Untuk melewati shutdown OS yang anggun selama penghentian paksa, sertakan `--skip-os-shutdown` opsi.

```
aws ec2 terminate-instances \
    --instance-ids i-1234567890abcdef0 \
```

```
    --force \
    --skip-os-shutdown
```

PowerShell

Untuk memaksakan penghentian sebuah instance

Gunakan Remove-EC2Instance cmdlet dan set `-Enforce` el ke. `true`

```
Remove-EC2Instance `
    -InstanceId i-1234567890abcdef0 `
    -Enforce $true
```

Untuk melewati shutdown OS yang anggun selama penghentian paksa, sertakan`-SkipOsShutdown $true`.

```
Remove-EC2Instance `
    -InstanceId i-1234567890abcdef0 `
    -Enforce $true `
    -SkipOsShutdown $true
```

## Instans yang dihentikan masih ditampilkan

Setelah Anda menghentikan suatu instans, instans tersebut akan tetap terlihat selama beberapa saat sebelum dihapus. Statusnya menunjukkan `terminated`. Jika entri tersebut tidak dihapus setelah beberapa jam, hubungi Dukungan.

## Kesalahan: Instans mungkin tidak dihentikan. Ubah atribut instance 'non ApiTermination aktifkan'

Jika Anda mencoba menghentikan instans dan mendapatkan pesan kesalahan `The instance i-1234567890abcdef0 may not be terminated. Modify its 'disableApiTermination' instance attribute`, pesan ini menunjukkan bahwa instans telah diaktifkan untuk perlindungan penghentian. Perlindungan penghentian mencegah instans dihentikan secara tidak sengaja.

Anda harus menonaktifkan perlindungan penghentian sebelum Anda dapat menghentikan instans.

Untuk informasi selengkapnya, lihat Ubah perlindungan penghentian instans.

## Instans diluncurkan atau dihentikan secara otomatis

Secara umum, perilaku berikut ini berarti Anda telah menggunakan Amazon EC2 Auto Scaling, Armada EC2, atau Armada Spot untuk menskalakan sumber daya komputasi Anda secara otomatis berdasarkan kriteria yang telah ditentukan:

- Anda menghentikan sebuah instans dan sebuah instans baru diluncurkan secara otomatis.
- Anda meluncurkan sebuah instans dan salah satu instans Anda dihentikan secara otomatis.
- Anda menghentikan sebuah instans dan instans tersebut akan terhenti, lalu instans baru akan diluncurkan secara otomatis.

Untuk menghentikan penskalaan otomatis, cari grup Penskalaan Otomatis atau armada yang meluncurkan instans dan atur kapasitasnya ke 0 atau hapus.

# Memecahkan masalah instans Amazon EC2 yang tidak dapat dijangkau

Informasi berikut dapat membantu Anda memecahkan masalah instans Amazon EC2 yang tidak dapat dijangkau. Anda dapat mengambil tangkapan layar atau mengakses output konsol untuk membantu mendiagnosis masalah dan menentukan apakah Anda harus me-reboot instance Anda. Untuk instans Windows yang tidak dapat dijangkau, pecahkan masalah dengan meninjau tangkapan layar yang dikembalikan oleh layanan.

Daftar Isi


- Boot ulang instans
- Output konsol instans
- Mengambil tangkapan layar instans yang tidak dapat dijangkau
- Tangkapan layar umum untuk memecahkan masalah instans Windows yang tidak dapat dijangkau
- Pemulihan instans saat komputer host gagal
- Instance muncul offline dan secara tak terduga di-reboot


## Boot ulang instans

Kemampuan untuk boot ulang instans yang tidak dapat dijangkau sangat berguna untuk pemecahan masalah dan manajemen instans umum.

Sama seperti Anda dapat mengatur ulang komputer dengan menekan tombol reset, Anda dapat mengatur ulang instans EC2 menggunakan konsol Amazon EC2, CLI, atau API. Untuk informasi selengkapnya, lihat Mulai ulang instans Amazon EC2 Anda.

## Output konsol instans

Output konsol adalah alat yang berguna untuk mendiagnosis masalah. Alat ini sangat berguna untuk memecahkan masalah kernel dan masalah konfigurasi layanan yang dapat menyebabkan sebuah instans diakhiri atau tidak dapat dijangkau sebelum daemon SSH-nya dapat dimulai.

- Instans Linux — Output konsol instance menampilkan output konsol yang tepat yang biasanya ditampilkan pada monitor fisik yang terpasang ke komputer. Output konsol menampilkan informasi singkat yang di-posting segera setelah status transisi instans (mulai, hentikan, mulai ulang, dan akhiri). Output yang di-posting tidak terus-menerus diperbarui; hanya jika kemungkinan besar nilainya paling tinggi.
- Instans Windows — Output konsol instance menyertakan tiga kesalahan log peristiwa sistem terakhir.

Hanya pemilik instans yang dapat mengakses output konsol.

Anda dapat mengambil output konsol serial terbaru selama siklus hidup instans. Opsi ini hanya didukung pada Nitro-based instance.

Console

Untuk mendapatkan output konsol

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans.
4. Pilih Tindakan, Memantau dan memecahkan masalah, Dapatkan sistem log.

AWS CLI

Untuk mendapatkan output konsol

Gunakan perintah get-console-output.

```
aws ec2 get-console-output --instance-id i-1234567890abcdef0
```

PowerShell

Untuk mendapatkan output konsol

Gunakan Get-EC2ConsoleOutput cmdlet.

```
Get-EC2ConsoleOutput -InstanceId i-1234567890abcdef0
```

## Mengambil tangkapan layar instans yang tidak dapat dijangkau

Jika Anda tidak dapat terhubung ke instans, Anda dapat mengambil tangkapan layar instans Anda dan melihatnya sebagai gambar. Gambar tersebut dapat memberikan visibilitas tentang status instans, dan memungkinkan pemecahan masalah yang lebih cepat.

Anda dapat menghasilkan tangkapan layar saat instans sedang berjalan atau setelah mengalami crash. Gambar dihasilkan dalam format JPG dan tidak lebih besar dari 100 kb. Tidak ada biaya transfer data untuk tangkapan layar ini.

Batasan

Fitur ini tidak didukung untuk hal-hal berikut:

- Instans bare metal (instans tipe `*.metal`)
- Instans menggunakan driver NVIDIA GRID
- Instans didukung oleh prosesor Arm-based Graviton
- Instans Windows aktif AWS Outposts
- Instans Windows di AWS Zona Lokal

Dukungan wilayah

Fitur ini tidak tersedia di Wilayah berikut:

- Asia Pasifik (Thailand)
- Meksiko (Tengah)
- GovCloud Daerah

Console

Untuk mendapatkan tangkapan layar suatu instans

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi kiri, pilih Instans.
3. Pilih instans yang akan diambil gambarnya.
4. Pilih Tindakan, Pantau dan atasi masalah, Dapatkan tangkapan layar instans.
5. Pilih Unduh, atau buka menu konteks (klik kanan) untuk gambar untuk mengunduh dan menyimpannya.

AWS CLI

Untuk mengambil tangkapan layar suatu instans

Gunakan perintah get-console-screenshot. Output-nya dikodekan base64.

```
aws ec2 get-console-screenshot --instance-id i-1234567890abcdef0
```

PowerShell

Untuk mengambil tangkapan layar suatu instans

Gunakan Get-EC2ConsoleScreenshot cmdlet. Output-nya dikodekan base64.

```
Get-EC2ConsoleScreenshot -InstanceId i-1234567890abcdef0
```

## Tangkapan layar umum untuk memecahkan masalah instans Windows yang tidak dapat dijangkau

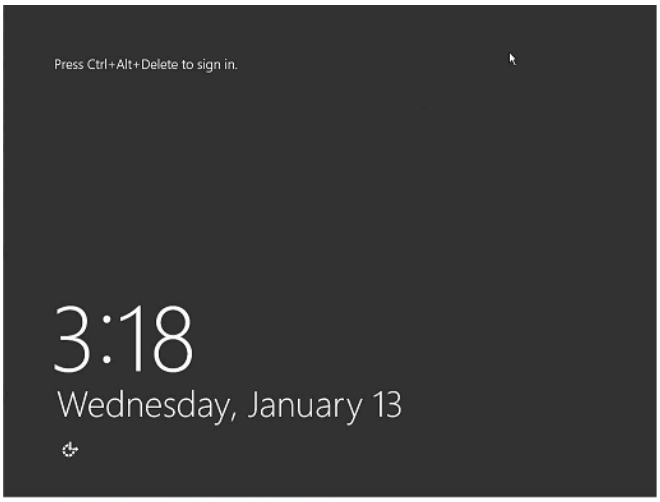
Anda dapat menggunakan informasi berikut untuk membantu memecahkan masalah instans yang tidak terjangkau berdasarkan tangkapan layar yang dikembalikan oleh layanan.

- Layar masuk (Ctrl + Alt + Delete)
- Layar konsol pemulihan
- Layar Windows boot manager

- Layar Sysprep
- Layar persiapan
- Layar Pembaruan Windows
- Chkdsk

## Layar masuk (Ctrl + Alt + Delete)

Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.







Jika sebuah instans tidak dapat dijangkau selama proses masuk, mungkin ada masalah dengan konfigurasi jaringan Anda atau Remote Desktop Services Windows. Instans juga bisa menjadi tidak responsif jika suatu proses menggunakan CPU dalam jumlah besar.

Konfigurasi jaringan

Gunakan informasi berikut untuk memverifikasi bahwa konfigurasi jaringan Anda AWS, Microsoft Windows, dan lokal (atau lokal) tidak memblokir akses ke instance.

AWS konfigurasi jaringan

| Konfigurasi | Verifikasi |
| --- | --- |
| Konfigurasi grup keamanan | Verifikasi bahwa port 3389 terbuka untuk grup keamanan Anda. Verifikasi bahwa Anda terhubung ke alamat IP publik yang benar. Jika |

| Konfigurasi | Verifikasi |
| --- | --- |
| | instance tidak dikaitkan dengan IP Elastic, IP publik berubah setelah instance stops/starts. Untuk informasi selengkapnya, lihat Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh. |
| Konfigurasi VPC (ACL Jaringan) | Verifikasi bahwa daftar kontrol akses (ACL) untuk Amazon VPC Anda tidak memblokir akses. Untuk mendapatkan informasi, lihat ACL Jaringan di Panduan Pengguna Amazon VPC. |
| Konfigurasi VPN | Jika Anda menghubungkan ke VPC menggunakan jaringan privat virtual (VPN), verifikasi konektivitas tunnel VPN. Untuk informasi selengkapnya, lihat Mem ecahkan AWS Masalah Klien VPN: Masalah konektivitas terowongan ke VPC. |

Konfigurasi jaringan Windows

| Konfigurasi | Verifikasi |
| --- | --- |
| Windows Firewall | Verifikasi bahwa Windows Firewall tidak memblokir koneksi ke instans Anda. Nonaktifk an Windows Firewall seperti yang dijelaskan pada poin 7 di bagian pemecahan masalah Desktop Jarak Jauh, Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh. |
| TCP/IP Konfigurasi lanjutan (Penggunaan IP statis) | Instans mungkin menjadi tidak responsif karena Anda mengonfigurasi alamat IP statis. Untuk VPC, buat antarmuka jaringan dan lampirkan ke instans. |

Konfigurasi Jaringan Lokal atau on-premise

Verifikasi bahwa konfigurasi jaringan lokal tidak memblokir akses. Coba hubungkan ke instans lain di VPC yang sama dengan instans tak terjangkau milik Anda. Jika Anda tidak dapat mengakses instans lain, bekerja sama dengan administrator jaringan lokal Anda untuk mencari tahu apakah kebijakan lokal membatasi akses.

Masalah Remote Desktop Services

Jika instans tidak dapat dijangkau selama proses masuk, mungkin ada masalah dengan Remote Desktop Services (RDS) pada instans.

> **Tip**
>
> Anda dapat menggunakan runbook `AWSSupport-TroubleshootRDP` untuk memeriksa dan mengubah berbagai pengaturan yang mungkin memengaruhi koneksi Remote Desktop Protocol (RDP). Untuk informasi selengkapnya, lihat `AWSSupport-TroubleshootRDP` di referensi buku runbook Otomatisasi AWS Systems Manager .

Konfigurasi Remote Desktop Services

| Konfigurasi | Verifikasi |
| --- | --- |
| RDS sedang berjalan | Verifikasi bahwa RDS sedang berjalan di instans. Hubungkan ke instans menggunakan snap-in Layanan Microsoft Managemen t Console (MMC) (`services.msc` ). Dalam daftar layanan, verifikasi bahwa Remote Desktop Services sedang Berjalan. Jika tidak, mulai dan atur tipe startup ke Otomatis. Jika Anda tidak dapat terhubung ke instans dengan menggunakan snap-in Layanan, lepaskan volume root dari instans, ambil snapshot volume atau buat AMI darinya, lampirkan volume asli ke instans lain di Zona Ketersediaan yang sama sebagai volume sekunder, dan ubah kunci registri Mulai. Setelah Anda selesai, lampirkan kembali volume root ke instans asli. |
| RDS diaktifkan | Bahkan jika layanan dimulai, RDS mungkin dinonaktifkan. Lepaskan volume root dari instans, ambil snapshot volume atau buat AMI darinya, lampirkan volume asli ke instans lain di Zona Ketersediaan yang sama sebagai volume sekunder, dan aktifkan layanan dengan mengubah kunci registri Server |

| Konfigurasi | Verifikasi |
| --- | --- |
| | Terminal seperti yang dijelaskan dalam Aktifkan Desktop Jarak Jauh pada instans EC2 dengan registri jarak jauh.<br><br>Setelah Anda selesai, lampirkan kembali volume root ke instans asli. |

Penggunaan CPU yang tinggi

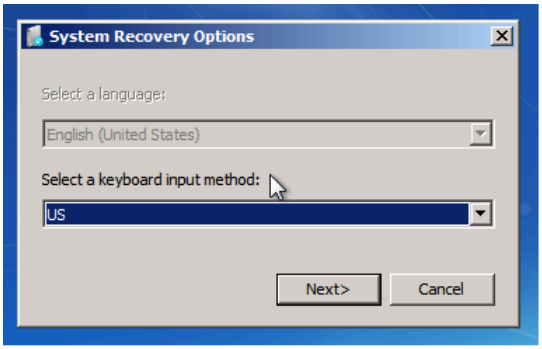
Periksa metrik CPUUtilization (Maximum) pada instans Anda dengan menggunakan Amazon. CloudWatch Jika CPUUtilization (Maksimum) menampilkan angka yang besar, tunggu hingga angka CPU turun dan coba sambungkan lagi. Penggunaan CPU yang tinggi dapat disebabkan oleh:

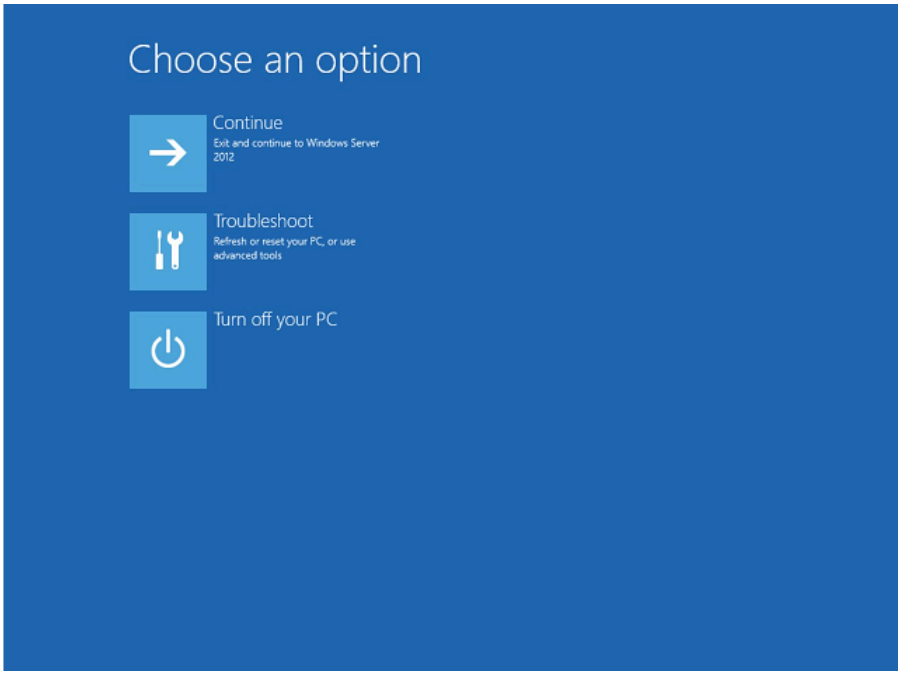
- Pembaruan Windows
- Pemindaian Perangkat Lunak Keamanan
- Skrip Startup Kustom
- Penjadwal Tugas

Untuk informasi selengkapnya, lihat Dapatkan Statistik untuk Sumber Daya Tertentu di Panduan CloudWatch Pengguna Amazon. Untuk kiat-kiat pemecahan masalah tambahan, lihat Penggunaan CPU tinggi segera setelah Windows dimulai (hanya instans Windows).

## Layar konsol pemulihan

Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.




Sistem operasi mungkin melakukan boot ke konsol Pemulihan dan terjebak di status ini jika `bootstatuspolicy` tidak diatur ke `ignoreallfailures`. Gunakan prosedur berikut untuk mengubah konfigurasi `bootstatuspolicy` ke `ignoreallfailures`.

Secara default, konfigurasi kebijakan untuk AMI Windows publik yang disediakan oleh AWS disetel `keignoreallfailures`.

1. Hentikan instans tak terjangkau.
2. Buat snapshot dari volume root. Volume root dilampirkan ke instans sebagai `/dev/sda1`.

   Lepaskan volume root dari instans tak terjangkau, ambil snapshot volume atau buat AMI darinya, dan lampirkan ke instans lain di Zona Ketersediaan yang sama dengan volume sekunder.

   > **Warning**
   >
   > Jika instans sementara dan asli Anda diluncurkan menggunakan AMI yang sama, Anda harus menyelesaikan langkah-langkah tambahan atau Anda tidak akan dapat melakukan boot instans asli setelah memulihkan volume root-nya karena tabrakan tanda tangan disk. Jika Anda harus membuat instans sementara menggunakan AMI yang sama, untuk menghindari tabrakan tanda tangan disk, selesaikan langkah-langkah di Tabrakan tanda tangan disk.

Atau, pilih AMI yang berbeda untuk instans sementara. Misalnya, jika instance asli menggunakan AMI untuk Windows Server 2016, luncurkan instance sementara menggunakan AMI untuk Windows Server 2019.

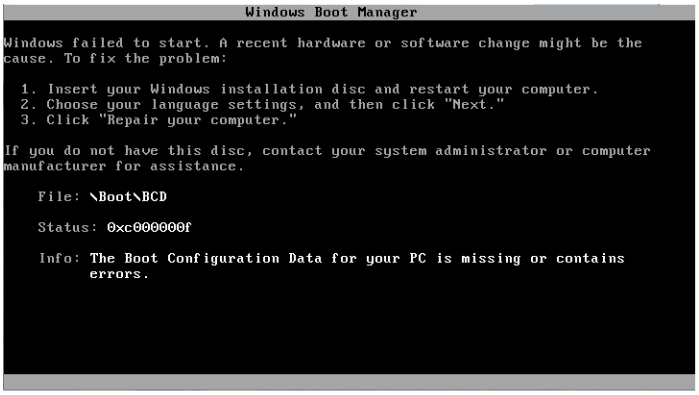
3. Masuk ke instans dan jalankan perintah berikut dari prompt perintah untuk mengubah konfigurasi `bootstatuspolicy` ke `ignoreallfailures`.

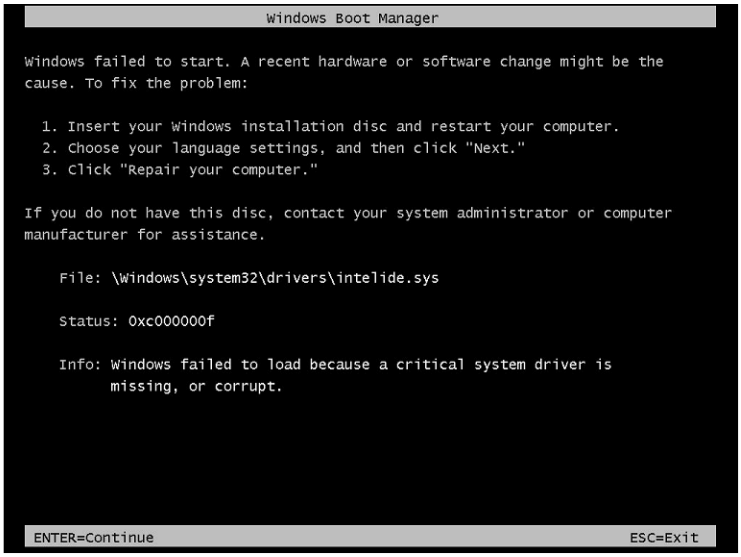
```
bcdedit /store Drive Letter:\boot\bcd /set {default} bootstatuspolicy
 ignoreallfailures
```

4. Lampirkan kembali volume ke instans tak terjangkau dan mulai instans lagi.

## Layar Windows boot manager

Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.

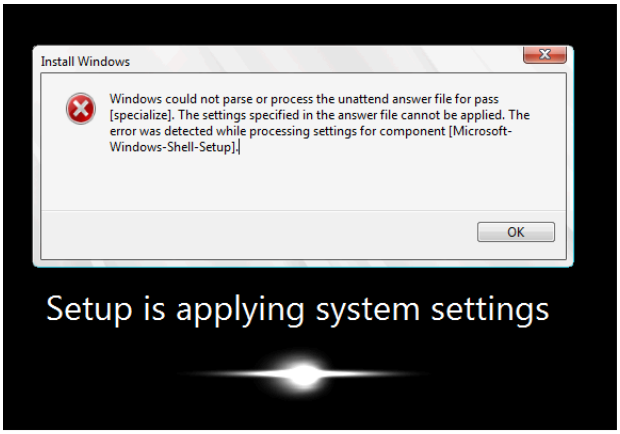


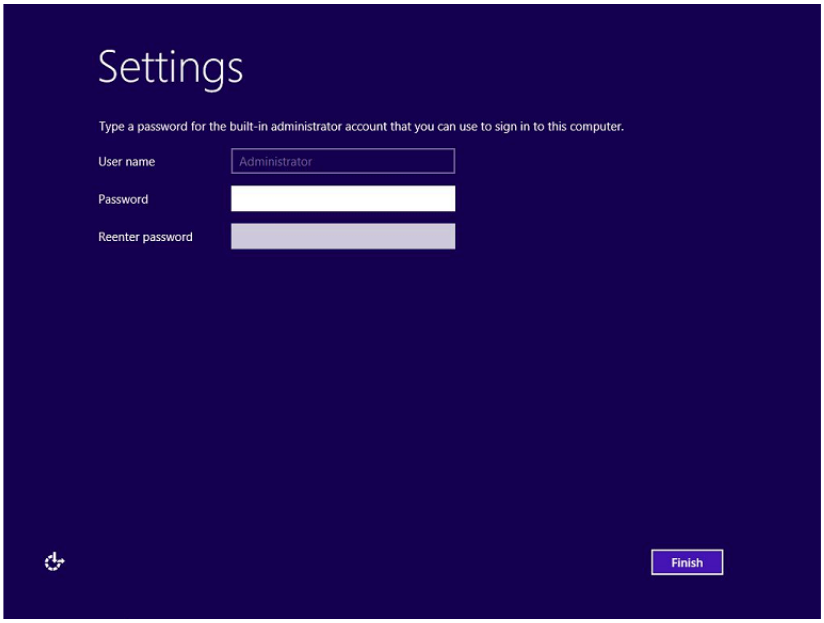




Sistem operasi mengalami korupsi fatal dalam file sistem registr and/or i. Saat instans terjebak dalam status ini, Anda harus memulihkan instans dari AMI cadangan terbaru atau meluncurkan instans pengganti. Jika Anda perlu mengakses data pada instans, lepaskan volume root apa pun dari instans tak terjangkau, ambil snapshot dari volume tersebut atau buat AMI darinya, dan lampirkan ke instans lain di Zona Ketersediaan yang sama dengan volume sekunder.

## Layar Sysprep

Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.

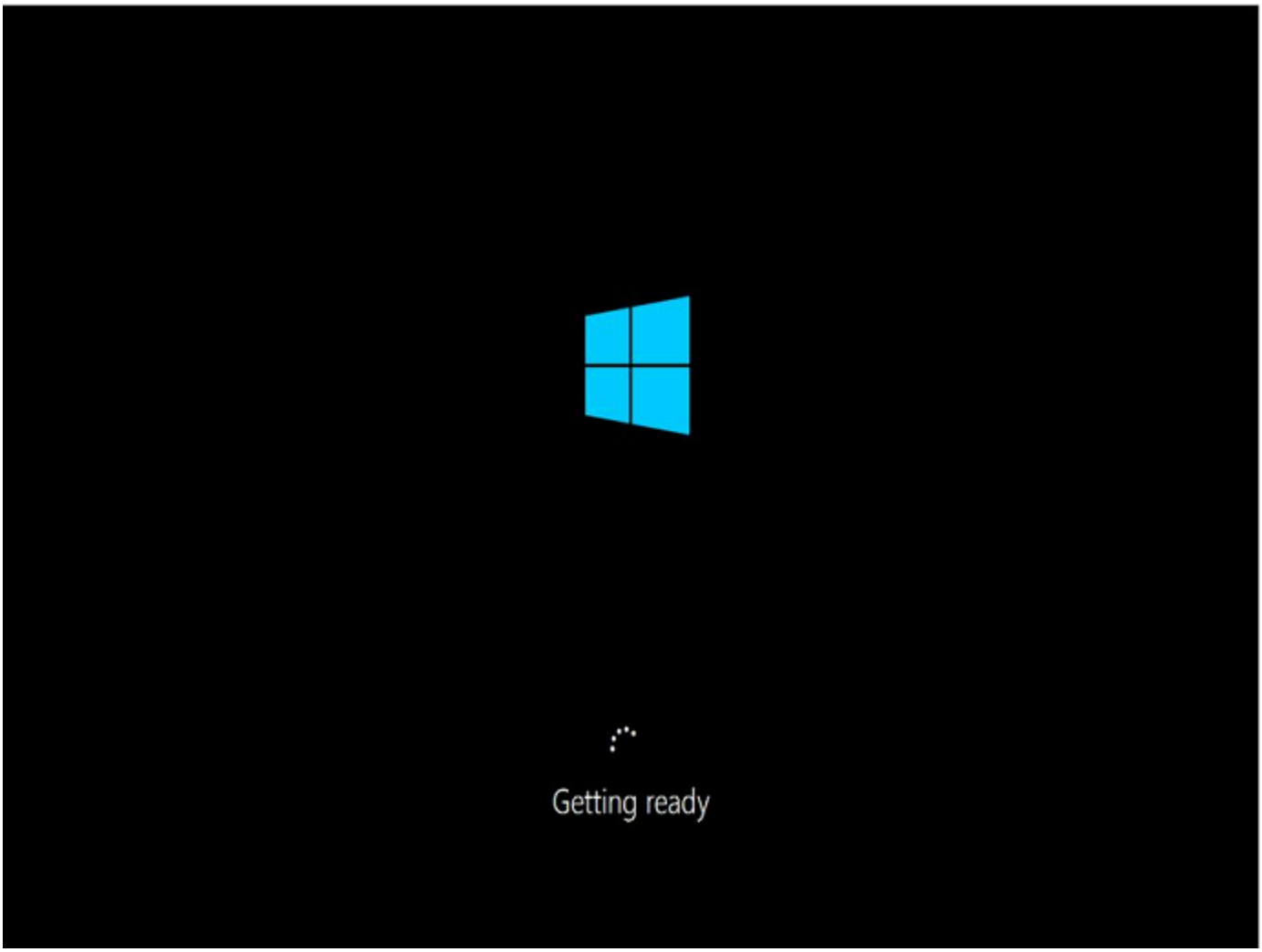



Anda mungkin melihat layar ini jika Anda tidak menggunakan Layanan EC2Config untuk memanggil Sysprep atau jika sistem operasi gagal saat menjalankan Sysprep. Anda dapat mengatur ulang kata sandi menggunakan EC2Rescue. Jika tidak, lihat Buat AMI Amazon EC2 menggunakan Windows Sysprep.

## Layar persiapan

Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.




Segarkan Layanan Tangkapan Layar Konsol Instans berulang kali untuk memverifikasi bahwa ring progres sedang berputar. Jika ring berputar, tunggu sistem operasi untuk memulai. Anda juga dapat memeriksa metrik CPUUtilization (Maximum) pada instans Anda dengan menggunakan Amazon CloudWatch untuk melihat apakah sistem operasi aktif. Jika ring progres tidak berputar, instans akan terjebak saat proses boot. Boot ulang instans. Jika melakukan boot ulang tidak menyelesaikan masalah, pulihkan instans dari AMI cadangan terbaru atau luncurkan instans pengganti. Jika Anda perlu mengakses data pada instans, lepaskan volume root dari instans tak terjangkau, ambil snapshot volume atau buat AMI darinya. Kemudian, lampirkan ke instans lain di Zona Ketersediaan yang sama dengan volume sekunder.

## Layar Pembaruan Windows

Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.

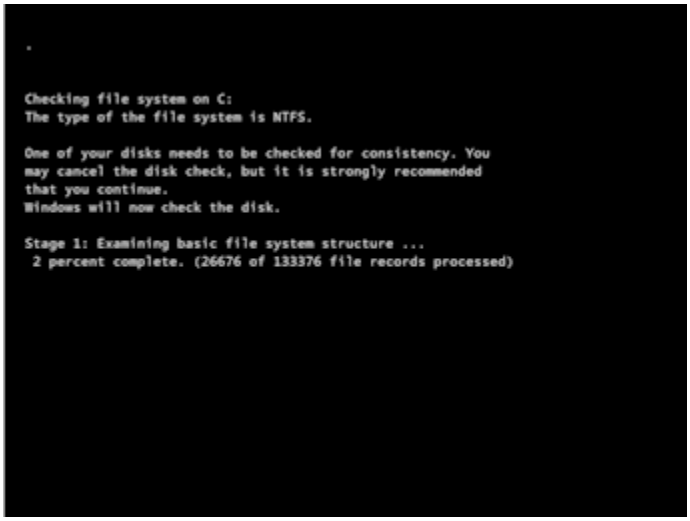



Proses Pembaruan Windows sedang memperbarui registri. Tunggu hingga pembaruan selesai. Jangan lakukan boot ulang atau penghentian instans karena ini dapat menyebabkan kerusakan data selama pembaruan.

> **Note**
>
> Proses Pembaruan Windows dapat menghabiskan sumber daya di server selama pembaruan. Jika Anda sering mengalami masalah ini, pertimbangkan untuk menggunakan tipe instans yang lebih cepat dan volume EBS yang lebih cepat.

## Chkdsk

Layanan Tangkapan Layar Konsol mengembalikan hal-hal berikut ini.




Windows menjalankan alat sistem chkdsk pada drive untuk memverifikasi integritas sistem file dan memperbaiki kesalahan sistem file logis. Tunggu hingga prosesnya selesai.

## Pemulihan instans saat komputer host gagal

Jika terdapat masalah yang tidak dapat dipulihkan dengan perangkat keras komputer host yang mendasarinya, AWS dapat menjadwalkan peristiwa penghentian instans. Anda akan terlebih dahulu diberi tahu tentang peristiwa tersebut melalui email.

Untuk memulihkan EBS-backed instans Amazon yang berjalan di komputer host yang gagal

1. Cadangkan semua data penting pada volume penyimpanan instans Anda ke Amazon EBS atau Amazon S3.
2. Hentikan instans.
3. Mulai instans.
4. Pulihkan setiap data penting.

Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.

Untuk memulihkan instance dengan volume root penyimpanan instance yang berjalan di komputer host yang gagal

1. Buat AMI dari instans.
2. Unggah gambar ke Amazon S3.
3. Cadangkan data penting ke Amazon EBS atau Amazon S3.
4. Akhiri instans.
5. Luncurkan instans baru dari AMI.
6. Pulihkan setiap data penting ke instans baru.

## Instance muncul offline dan secara tak terduga di-reboot

Jika instans Anda tampaknya offline dan kemudian di-boot ulang secara tidak terduga, instans mungkin telah mengalami pemulihan instans otomatis. Ini terjadi ketika AWS mendeteksi bahwa instance tidak tersedia karena masalah perangkat keras atau perangkat lunak yang mendasarinya, dan pemulihan otomatis yang disederhanakan atau pemulihan berbasis CloudWatch tindakan diaktifkan pada instans.

Selama proses pemulihan, mencoba AWS mengembalikan ketersediaan instance dengan memigrasinya ke perangkat keras yang berbeda. Untuk memverifikasi apakah pemulihan instans otomatis terjadi untuk instans Anda, lihatVerifikasi apakah pemulihan instans otomatis terjadi.

> **Note**
>
> Jika beban kerja atau aplikasi Anda tidak responsif, periksa apakah itu berjalan pada instans. Jika tidak, mulailah secara manual. Untuk mencegah masalah ini di masa mendatang, terapkan rencana pemulihan untuk memastikan beban kerja atau aplikasi berfungsi dengan benar setelah pemulihan instans.

# Memecahkan masalah saat menyambung ke instans Amazon EC2 Linux

Informasi berikut dan kesalahan umum dapat membantu Anda memecahkan masalah saat terhubung ke instans Linux Anda.

Masalah koneksi

- Penyebab umum masalah koneksi
- Kesalahan saat menghubungkan instans Anda: Waktu koneksi habis
- Kesalahan: tidak dapat memuat kunci ... Mengharapkan: KUNCI PRIVAT APA PUN
- Kesalahan: Kunci pengguna tidak dikenali oleh server
- Kesalahan: Izin ditolak atau koneksi ditutup oleh [instans] port 22
- Kesalahan: File kunci privat yang tidak dilindungi
- Kesalahan: Kunci privat harus dimulai dengan “-----BEGIN RSA PRIVATE KEY-----” dan diakhiri dengan “-----END RSA PRIVATE KEY-----”
- Kesalahan: Verifikasi kunci host gagal
- Kesalahan: Server menolak kunci kami atau Tidak ada metode otentikasi yang didukung tersedia
- Tidak dapat melakukan ping pada instans
- Kesalahan: Server menutup koneksi jaringan secara tidak terduga
- Kesalahan: Validasi kunci host gagal untuk EC2 Instance Connect
- Tidak dapat terhubung ke instans Ubuntu menggunakan EC2 Instance Connect

- Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance saya?

## Penyebab umum masalah koneksi

Sebaiknya Anda mulai memecahkan masalah koneksi instans dengan memverifikasi bahwa Anda telah melakukan tugas-tugas berikut secara akurat.

Verifikasi nama pengguna untuk instans Anda

Anda dapat terhubung ke instans menggunakan nama pengguna untuk akun pengguna Anda atau nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instans Anda.

- Dapatkan nama pengguna untuk akun pengguna Anda.

  Untuk informasi selengkapnya tentang cara membuat akun pengguna, lihat Mengelola pengguna sistem di instans Amazon EC2 Linux Anda.

- Dapatkan nama pengguna default untuk AMI yang Anda gunakan untuk meluncurkan instans Anda.

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
|---|---|
| Amazon Linux | `ec2-user` |
| CentOS | `centos` atau `ec2-user` |
| Debian | `admin` |
| Fedora | `fedora` atau `ec2-user` |
| FreeBSD | `ec2-user` |
| RHEL | `ec2-user` atau `root` |
| SEGAR | `ec2-user` atau `root` |
| Ubuntu | `ubuntu` |
| Oracle | `ec2-user` |
| Bitnami | `bitnami` |

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
| --- | --- |
| Linux Rocky | `rocky` |
| Lainnya | Periksa dengan penyedia AMI |

Verifikasi bahwa aturan grup keamanan mengizinkan lalu lintas

Pastikan grup keamanan yang terkait dengan instans Anda mengizinkan lalu lintas SSH masuk dari alamat IP Anda. Grup keamanan default untuk VPC tidak mengizinkan lalu lintas SSH masuk secara default. Grup keamanan yang dibuat oleh wizard peluncuran instans mengaktifkan lalu lintas SSH secara default. Untuk langkah-langkah guna menambahkan aturan lalu lintas SSH masuk ke instans Linux Anda, lihat. Aturan-aturan untuk terhubung ke instans dari komputer Anda Untuk langkah-langkah verifikasi, lihat Kesalahan saat menghubungkan instans Anda: Waktu koneksi habis.

Verifikasi bahwa instans Anda sudah siap

Setelah Anda meluncurkan instance, perlu beberapa menit agar instans siap menerima permintaan koneksi. Periksa instans Anda untuk memastikan ia berfungsi dan telah melewati pemeriksaan status.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans, kemudian pilih instans Anda.
3. Verifikasi hal berikut:

    a. Di kolom Status instans, verifikasi bahwa instans Anda berada dalam status `running`.

    b. Di kolom Pemeriksaan status, verifikasi bahwa instance Anda telah lulus semua pemeriksaan status.

Verifikasi bahwa Anda telah memenuhi semua prasyarat untuk terhubung

Pastikan Anda memiliki semua informasi yang Anda butuhkan untuk terhubung. Untuk informasi selengkapnya, lihat Hubungkan ke instans Linux Anda, menggunakan SSH..

Hubungkan dari Linux atau macOS X

Jika sistem operasi komputer lokal Anda adalah Linux atau macOS X, periksa berikut ini untuk prasyarat khusus untuk menghubungkan ke instance Linux:

- Klien SSH

- EC2 Instance Connect
- AWS Systems Manager Manajer Sesi

Hubungkan dari Windows

Jika sistem operasi komputer lokal Anda adalah Windows, periksa berikut ini untuk prasyarat khusus untuk menghubungkan ke instance Linux:

- OpenSSH
- PuTTY
- AWS Systems Manager Manajer Sesi
- Subsistem Windows untuk Linux

Periksa apakah instance tersebut merupakan instance terkelola

User-initiated koneksi ke instans terkelola tidak diperbolehkan. Untuk menentukan apakah instance dikelola, cari bidang Ter kelola untuk instance tersebut. Jika nilainya benar, itu adalah instance terkelola. Untuk informasi selengkapnya, lihat Instans terkelola Amazon EC2.

## Kesalahan saat menghubungkan instans Anda: Waktu koneksi habis

Jika Anda mencoba terhubung ke instans Anda dan mendapatkan pesan kesalahan `Network error: Connection timed out` atau`Error connecting to [instance], reason: -> Connection timed out: connect`, coba yang berikut ini:

Periksa aturan grup keamanan Anda.

Anda memerlukan aturan grup keamanan yang mengizinkan lalu lintas masuk dari alamat IPv4 publik di port yang sesuai.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans, kemudian pilih instans Anda.
3. Pada tab Keamanan di bagian bawah halaman konsol, di bawah Aturan Masuk, periksa daftar aturan yang memengaruhi instans terpilih. Verifikasi bahwa ada aturan yang memungkinkan lalu lintas dari komputer lokal Anda ke port 22 (SSH).

   Jika grup keamanan Anda tidak memiliki aturan yang mengizinkan lalu lintas masuk dari komputer lokal Anda, tambahkan aturan ke grup keamanan Anda. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.

4. Untuk aturan yang memungkinkan lalu lintas masuk, periksa bidang Sumber. Jika nilainya adalah alamat IP tunggal, dan jika alamat IP tidak statis, alamat IP baru akan ditetapkan setiap kali Anda memulai ulang komputer Anda. Ini akan mengakibatkan aturan tidak menyertakan lalu lintas alamat IP komputer Anda. Alamat IP mungkin tidak statis jika komputer Anda berada di jaringan perusahaan, atau Anda terhubung melalui penyedia layanan internet (ISP), atau alamat IP komputer Anda dinamis dan berubah setiap kali Anda memulai ulang komputer Anda. Untuk memastikan bahwa aturan grup keamanan Anda mengizinkan lalu lintas masuk dari komputer lokal Anda, alih-alih menentukan satu alamat IP untuk Sumber, lebih baik tentukan rentang alamat IP yang digunakan oleh komputer klien Anda.

   Untuk informasi selengkapnya tentang aturan grup keamanan, lihat Aturan grup keamanan di Panduan Pengguna Amazon VPC.

Periksa tabel rute untuk subnet.

Anda memerlukan rute yang mengirimkan semua lalu lintas yang ditujukan di luar VPC ke gateway internet untuk VPC.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans, kemudian pilih instans Anda.
3. Pada tab Jaringan, catat nilai untuk ID VPC dan ID subnet.
4. Buka konsol Amazon VPC di https://console.aws.amazon.com/vpc/.
5. Di panel navigasi, pilih Gateway Internet. Verifikasi bahwa ada gateway internet yang dilampirkan ke VPC Anda. Jika tidak ada, pilih Buat gateway internet, masukkan nama untuk gateway internet, dan pilih Buat gateway internet. Kemudian, untuk gateway internet yang Anda buat, pilih Tindakan, Lampirkan ke VPC, pilih VPC Anda, lalu pilih Lampirkan gateway internet untuk melampirkannya ke VPC anda.
6. Di panel navigasi, pilih Subnet, lalu pilih subnet Anda.
7. Di tab Tabel Rute, verifikasi bahwa ada rute dengan `0.0.0.0/0` sebagai tujuan dan gateway internet untuk VPC Anda sebagai target. Jika Anda terhubung ke instans Anda menggunakan alamat IPv6, pastikan ada rute untuk semua lalu lintas IPv6 (`::/0`) yang mengarah ke gateway internet. Jika tidak, lakukan tindakan berikut:

   a. Pilih ID tabel rute (rtb-xxxxxxxx) untuk menavigasi ke tabel rute.

b. Di tab Rute, pilih Edit rute. Pilih Tambahkan rute, gunakan `0.0.0.0/0` sebagai tujuan, dan gateway internet sebagai target. Untuk IPv6, pilih Tambahkan rute, gunakan `::/0` sebagai tujuan, dan gateway internet sebagai target.

c. Pilih Simpan rute.

Periksa jaringan daftar kontrol akses (ACL) untuk subnet.

ACL jaringan harus mengizinkan lalu lintas SSH masuk dari alamat IP lokal Anda pada port 22. Hal ini juga mengizinkan lalu lintas keluar ke port sementara (1024-65535).

1. Buka konsol Amazon VPC di https://console.aws.amazon.com/vpc/.
2. Di panel navigasi, pilih Pengguna.
3. Pilih subnet Anda.
4. Pada tab ACL Jaringan, untuk Aturan masuk, verifikasi bahwa aturan mengizinkan lalu lintas masuk dari komputer Anda di port yang diperlukan. Jika tidak, hapus atau ubah aturan yang memblokir lalu lintas.
5. Untuk Aturan keluar, verifikasi bahwa aturan mengizinkan lalu lintas keluar ke komputer Anda pada port sementara. Jika tidak, hapus atau ubah aturan yang memblokir lalu lintas.

Jika komputer Anda berada di jaringan perusahaan

Tanyakan administrator jaringan Anda apakah firewall internal memungkinkan lalu lintas masuk dan keluar dari komputer Anda pada port 22.

Jika Anda memiliki firewall di komputer Anda, verifikasi bahwa itu memungkinkan lalu lintas masuk dan keluar dari komputer Anda di port 22.

Periksa apakah instans Anda memiliki alamat IPv4 publik atau tidak.

Jika tidak, Anda dapat mengaitkan alamat IP Elastis dengan instans Anda. Untuk informasi selengkapnya, lihat Alamat Elastic IP.

Periksa beban CPU pada instans Anda; server mungkin kelebihan beban.

AWS secara otomatis menyediakan data seperti CloudWatch metrik Amazon dan status instans, yang dapat Anda gunakan untuk melihat berapa banyak beban CPU pada instans Anda dan, jika perlu, menyesuaikan cara penanganan beban Anda. Untuk informasi selengkapnya, lihat Memantau instans Anda menggunakan CloudWatch.

- Jika beban Anda bervariasi, Anda dapat menaikkan atau menurunkan skala secara otomatis menggunakan Auto Scaling dan Elastic Load Balancing.
- Jika beban terus bertambah, Anda dapat beralih ke tipe instans yang lebih besar. Untuk informasi selengkapnya, lihat Perubahan jenis instans Amazon EC2.

Untuk terhubung ke instans Anda menggunakan alamat IPv6, periksa hal berikut ini:

- Subnet Anda harus terkait dengan tabel rute yang memiliki rute untuk lalu lintas IPv6 (`::/0`) ke gateway internet.
- Aturan grup keamanan Anda harus mengizinkan lalu lintas masuk dari alamat IPv6 lokal Anda di port 22.
- Aturan ACL jaringan Anda harus mengizinkan lalu lintas IPv6 masuk dan keluar.
- Jika Anda meluncurkan instans dari versi AMI yang lebih lama, instans mungkin tidak dikonfigurasi untuk DHCPv6 (alamat IPv6 tidak dikenali secara otomatis pada antarmuka jaringan). Untuk informasi selengkapnya, lihat Konfigurasi IPv6 di instans Anda di Panduan Pengguna Amazon VPC.
- Komputer lokal Anda harus mempunyai alamat IPv6 dan harus dikonfigurasi untuk menggunakan IPv6.

## Kesalahan: tidak dapat memuat kunci ... Mengharapkan: KUNCI PRIVAT APA PUN

Jika Anda mencoba untuk terhubung ke instans Anda dan mendapatkan pesan kesalahan, `unable to load key ... Expecting: ANY PRIVATE KEY`, file tempat kunci privat disimpan tidak dikonfigurasi dengan benar. Jika file kunci privat berakhir dengan `.pem`, ia mungkin masih dikonfigurasi dengan salah. Kemungkinan penyebab file kunci privat yang tidak dikonfigurasi dengan benar adalah sertifikat yang hilang.

Jika file kunci privat tidak dikonfigurasi dengan benar, ikuti langkah-langkah berikut ini untuk mengatasi kesalahan

1. Buat pasangan kunci baru. Untuk informasi selengkapnya, lihat Membuat pasangan kunci menggunakan Amazon EC2.

> **Note**
>
> Sebagai gantinya, Anda dapat membuat pasangan kunci baru menggunakan alat pihak ketiga. Untuk informasi selengkapnya, lihat Membuat pasangan kunci menggunakan alat pihak ketiga dan mengimpor kunci publik ke Amazon EC2.

2. Tambahkan pasangan kunci baru ke instans Anda. Untuk informasi selengkapnya, lihat Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance saya?.
3. Hubungkan ke instans Anda menggunakan pasangan kunci baru.

## Kesalahan: Kunci pengguna tidak dikenali oleh server

Jika Anda menggunakan SSH untuk terhubung ke instans Anda

- Gunakan `ssh -vvv` untuk mendapatkan informasi debug tiga kali lipat saat menghubungkan:

```
ssh -vvv -i path/key-pair-name.pem instance-user-
name@ec2-203-0-113-25.compute-1.amazonaws.com
```

Output contoh berikut menunjukkan hal-hal yang mungkin Anda lihat jika mencoba untuk terhubung ke instans menggunakan kunci yang tidak dikenali oleh server:

```
open/ANT/myusername/.ssh/known_hosts).
debug2: bits set: 504/1024
debug1: ssh_rsa_verify: signature correct
debug2: kex_derive_keys
debug2: set_newkeys: mode 1
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug2: set_newkeys: mode 0
debug1: SSH2_MSG_NEWKEYS received
debug1: Roaming not allowed by server
debug1: SSH2_MSG_SERVICE_REQUEST sent
debug2: service_accept: ssh-userauth
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug2: key: boguspem.pem ((nil))
debug1: Authentications that can continue: publickey
debug3: start over, passed a different list publickey
```

```
debug3: preferred gssapi-keyex,gssapi-with-mic,publickey,keyboard-
interactive,password
debug3: authmethod_lookup publickey
debug3: remaining preferred: keyboard-interactive,password
debug3: authmethod_is_enabled publickey
debug1: Next authentication method: publickey
debug1: Trying private key: boguspem.pem
debug1: read PEM private key done: type RSA
debug3: sign_and_send_pubkey: RSA 9c:4c:bc:0c:d0:5c:c7:92:6c:8e:9b:16:e4:43:d8:b2
debug2: we sent a publickey packet, wait for reply
debug1: Authentications that can continue: publickey
debug2: we did not send a packet, disable method
debug1: No more authentication methods to try.
Permission denied (publickey).
```

Jika Anda menggunakan PuTTY untuk terhubung ke instans Anda

- Verifikasi bahwa file kunci privat (.pem) Anda telah diubah dengan sesuai ke format yang dikenali oleh PuTTY (.ppk). Untuk informasi selengkapnya tentang cara mengubah kunci privat, lihat Hubungkan ke instance Linux Anda menggunakan PuTTY.

  > **Note**
  >
  > Di dalam PuTTYgen, muat file kunci privat Anda dan pilih Simpan Kunci Privat alih-alih Buat.

- Verifikasi bahwa Anda terhubung dengan nama pengguna yang sesuai untuk AMI Anda. Masukkan nama pengguna di kotak Nama Host di jendela Konfigurasi PuTTY.

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
|---|---|
| Amazon Linux | `ec2-user` |
| CentOS | `centos` atau `ec2-user` |
| Debian | `admin` |
| Fedora | `fedora` atau `ec2-user` |
| FreeBSD | `ec2-user` |

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
| --- | --- |
| RHEL | `ec2-user` atau `root` |
| SEGAR | `ec2-user` atau `root` |
| Ubuntu | `ubuntu` |
| Oracle | `ec2-user` |
| Bitnami | `bitnami` |
| Linux Rocky | `rocky` |
| Lainnya | Periksa dengan penyedia AMI |

- Verifikasi bahwa Anda memiliki aturan grup keamanan masuk untuk mengizinkan lalu lintas masuk ke port yang sesuai. Untuk informasi selengkapnya, lihat Aturan-aturan untuk terhubung ke instans dari komputer Anda.

## Kesalahan: Izin ditolak atau koneksi ditutup oleh [instans] port 22

Jika Anda menyambung ke instance menggunakan SSH dan mendapatkan salah satu kesalahan berikut`Host key not found in [directory]`,`Permission denied (publickey)`,`Authentication failed, permission denied`,, atau`Connection closed by [instance] port 22`, verifikasi bahwa Anda terhubung dengan nama pengguna yang sesuai untuk AMI Anda dan bahwa Anda telah menentukan kunci pribadi yang tepat (`.pem`)file untuk instance Anda.

Nama pengguna yang sesuai adalah sebagai berikut:

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
| --- | --- |
| Amazon Linux | `ec2-user` |
| CentOS | `centos` atau `ec2-user` |
| Debian | `admin` |
| Fedora | `fedora` atau `ec2-user` |

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
|---|---|
| FreeBSD | `ec2-user` |
| RHEL | `ec2-user` atau `root` |
| SEGAR | `ec2-user` atau `root` |
| Ubuntu | `ubuntu` |
| Oracle | `ec2-user` |
| Bitnami | `bitnami` |
| Linux Rocky | `rocky` |
| Lainnya | Periksa dengan penyedia AMI |

Misalnya, untuk menggunakan klien SSH agar terhubung ke instans Amazon Linux, gunakan perintah berikut:

```
ssh -i /path/key-pair-name.pem instance-user-
name@ec2-203-0-113-25.compute-1.amazonaws.com
```

Konfirmasi bahwa Anda menggunakan file kunci privat yang sesuai dengan pasangan kunci, yang Anda pilih saat meluncurkan instans.

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans, lalu pilih instans Anda.
3. Pada tab Detail, di bawah Detail instans, verifikasi nilai Nama pasangan kunci.
4. Jika Anda tidak menentukan pasangan kunci saat meluncurkan instans, Anda dapat mengakhiri instans dan meluncurkan instans baru, untuk memastikan bahwa Anda telah menentukan pasangan kunci. Jika ini adalah instans yang Anda gunakan tetapi Anda tidak lagi memiliki file `.pem` untuk pasangan kunci, Anda bisa mengganti pasangan kunci dengan yang baru. Untuk informasi selengkapnya, lihat Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance saya?.

Jika Anda membuat pasangan kunci sendiri, pastikan bahwa pembuat kunci Anda diatur untuk membuat kunci RSA. Kunci DSA tidak diterima.

Jika Anda mendapatkan kesalahan `Permission denied (publickey)` dan tidak ada satu pun di atas yang berlaku (misalnya, Anda dapat terhubung sebelumnya), izin pada direktori beranda instans Anda mungkin telah diubah. Izin untuk /home/*instance-user-name*/.ssh/authorized_keys harus dibatasi untuk pemilik saja.

Untuk memverifikasi izin pada instans Anda

1. Hentikan instans Anda dan lepaskan volume root. Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.
2. Luncurkan instans sementara di Zona Ketersediaan yang sama dengan instans Anda saat ini (gunakan AMI yang serupa atau sama dengan yang digunakan untuk instans Anda saat ini), dan lampirkan volume root ke instans sementara.
3. Hubungkan ke instans sementara, buat titik pemasangan, dan pasang volume yang Anda lampirkan.
4. Dari instans sementara, periksa izin direktori /home/*instance-user-name*/ dari volume yang dilampirkan. Jika perlu, sesuaikan izin sebagai berikut:

```
[ec2-user ~]$ chmod 600 mount_point/home/instance-user-name/.ssh/authorized_keys
```

```
[ec2-user ~]$ chmod 700 mount_point/home/instance-user-name/.ssh
```

```
[ec2-user ~]$ chmod 700 mount_point/home/instance-user-name
```

5. Copot volume, lepas dari instans sementara, dan lampirkan kembali ke instans asli. Pastikan Anda menentukan nama perangkat yang benar untuk volume root; misalnya, /dev/xvda.
6. Mulai instans Anda. Jika Anda tidak lagi membutuhkan instans sementara, Anda dapat mengakhirinya.

## Kesalahan: File kunci privat yang tidak dilindungi

File kunci privat Anda harus dilindungi dari operasi baca dan tulis dari pengguna lain. Jika kunci privat Anda dapat dibaca atau ditulis oleh siapa pun kecuali Anda, SSH akan mengabaikan kunci Anda dan Anda akan melihat pesan peringatan berikut di bawah ini.

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions 0777 for '.ssh/my_private_key.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
bad permissions: ignore key: .ssh/my_private_key.pem
Permission denied (publickey).
```

Jika Anda melihat pesan serupa saat mencoba masuk ke instans, periksa baris pertama pesan kesalahan untuk memverifikasi bahwa Anda menggunakan kunci publik yang benar untuk instans Anda. Contoh di atas menggunakan kunci privat `.ssh/my_private_key.pem` dengan izin file `0777`, yang memungkinkan siapa pun untuk membaca atau menulis file ini. Tingkat izin ini sangat tidak aman, sehingga SSH mengabaikan kunci ini.

Jika Anda terhubung dari macOS atau Linux, jalankan perintah berikut untuk memperbaiki kesalahan ini, ganti jalur untuk file kunci privat Anda.

```
[ec2-user ~]$ chmod 0400 .ssh/my_private_key.pem
```

Jika Anda menghubungkan ke instance Linux dari Windows, lakukan langkah-langkah berikut di komputer lokal Anda.

1. Navigasikan ke file .pem Anda.
2. Buka menu konteks (klik kanan) untuk file.pem dan pilih Proper  ties.
3. Pilih tab Keamanan.
4. Pilih Lanjutan.
5. Verifikasi bahwa Anda adalah pemilik file. Jika tidak, ubah pemilik ke nama pengguna Anda.
6. Pilih Nonaktifkan warisan dan Hapus semua izin yang diwariskan dari objek ini.
7. Pilih Tambahkan, Pilih pengguna utama, masukkan nama pengguna, lalu pilih OKE.
8. Dari jendela Entri izin, berikan izin Baca dan pilih OKE.
9. Klik Terapkan untuk memastikan semua pengaturan disimpan.
10. Pilih OKE untuk menutup jendela Pengaturan Keamanan Lanjutan.
11. Pilih OKE untuk menutup jendela Properti.
12. Anda harus dapat terhubung ke instance Linux Anda dari Windows menggunakan SSH.

Dari prompt perintah Windows, jalankan perintah berikut.

1. Dari perintah prompt, navigasikan ke jalur lokasi file .pem Anda.
2. Jalankan perintah berikut untuk mengatur ulang dan menghapus izin eksplisit:

```
icacls.exe $path /reset
```

3. Jalankan perintah berikut untuk memberikan izin Baca kepada pengguna saat ini:

```
icacls.exe $path /GRANT:R "$($env:USERNAME):(R)"
```

4. Jalankan perintah berikut untuk menonaktifkan warisan dan menghapus izin yang diwariskan.

```
icacls.exe $path /inheritance:r
```

5. Anda harus dapat terhubung ke instance Linux Anda dari Windows menggunakan SSH.

## Kesalahan: Kunci privat harus dimulai dengan “-----BEGIN RSA PRIVATE KEY-----” dan diakhiri dengan “-----END RSA PRIVATE KEY-----”

Jika Anda menggunakan alat pihak ke tiga, seperti ssh-keygen, untuk membuat pasangan kunci RSA, alat ini menghasilkan kunci privat dalam format kunci OpenSSH. Saat Anda terhubung dengan instans, jika Anda menggunakan kunci privat dalam format OpenSSH untuk mendekripsi kata sandi, Anda akan menerima kesalahan `Private key must begin with "-----BEGIN RSA PRIVATE KEY-----" and end with "-----END RSA PRIVATE KEY-----"`.

Untuk mengatasi kesalahan tersebut, kunci privat harus dalam format PEM. Gunakan perintah berikut untuk membuat kunci privat dalam format PEM:

```
ssh-keygen -m PEM
```

## Kesalahan: Verifikasi kunci host gagal

Kesalahan ini terjadi jika ada ketidakcocokan antara kunci host yang disimpan pada instance dalam `known_hosts` file dan pada klien. Misalnya, ketidakcocokan dapat terjadi jika Anda menyambung ke instance menggunakan satu alamat IP publik, dan kemudian mencoba menyambungkannya lagi menggunakan alamat IP publik yang berbeda. Hal ini dapat terjadi setelah Anda menambahkan atau menghapus alamat IP Elastic, karena hal itu mengubah alamat IP publik dari sebuah instance.

Untuk mengatasi kesalahan ini, mulailah dengan mengonfirmasi bahwa ada perubahan yang diharapkan pada kunci host atau konfigurasi jaringan instance. Sebelum Anda terhubung ke instance, Anda mungkin juga ingin memverifikasi sidik jari host. Setelah Anda terhubung ke instance, Anda dapat menghapus kunci host lama dari `known_hosts` file. Untuk petunjuk, lihat dokumentasi untuk distribusi Linux yang digunakan pada instance Anda.

## Kesalahan: Server menolak kunci kami atau Tidak ada metode otentikasi yang didukung tersedia

Jika Anda menggunakan PuTTY untuk menyambung ke instance Anda dan mendapatkan salah satu dari kesalahan berikut, Kes alahan: Server menolak kunci kami atau Kes alahan: Tidak ada metode otentikasi yang didukung tersedia, verifikasi bahwa Anda terhubung dengan nama pengguna yang sesuai untuk AMI Anda. Ketik nama pengguna di Nama pengguna di jendela Konfigur asi PuTTY.

Nama pengguna yang sesuai adalah sebagai berikut:

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
|---|---|
| Amazon Linux | `ec2-user` |
| CentOS | `centos` atau `ec2-user` |
| Debian | `admin` |
| Fedora | `fedora` atau `ec2-user` |
| FreeBSD | `ec2-user` |
| RHEL | `ec2-user` atau `root` |
| SEGAR | `ec2-user` atau `root` |
| Ubuntu | `ubuntu` |
| Oracle | `ec2-user` |
| Bitnami | `bitnami` |
| Linux Rocky | `rocky` |

| AMI digunakan untuk meluncurkan instans | Nama pengguna default |
| --- | --- |
| Lainnya | Periksa dengan penyedia AMI |

Anda juga harus memverifikasi bahwa:

- Anda menggunakan PuTTY versi terbaru. Untuk informasi selengkapnya, lihat halaman web PuTTY.
- File kunci privat (.pem) Anda telah diubah dengan sesuai ke format yang dikenali oleh PuTTY (.ppk). Untuk informasi selengkapnya tentang cara mengubah kunci privat, lihat Hubungkan ke instance Linux Anda menggunakan PuTTY.

## Tidak dapat melakukan ping pada instans

Perintah `ping` adalah tipe lalu lintas ICMP — jika Anda tidak dapat melakukan ping pada instans Anda, pastikan bahwa aturan grup keamanan masuk Anda mengizinkan lalu lintas ICMP untuk pesan `Echo Request` dari semua sumber, atau dari komputer atau instans yang mengeluarkan perintah.

Jika Anda tidak dapat mengeluarkan perintah `ping` dari instans, pastikan bahwa aturan grup keamanan keluar Anda mengizinkan lalu lintas ICMP untuk pesan `Echo Request` ke semua tujuan, atau ke host yang Anda coba ping.

Perintah `Ping` juga dapat diblokir oleh firewall atau waktunya habis karena masalah latensi jaringan atau perangkat keras. Anda harus berkonsultasi dengan jaringan lokal atau administrator sistem untuk bantuan pemecahan masalah lebih lanjut.

## Kesalahan: Server menutup koneksi jaringan secara tidak terduga

Jika Anda terhubung ke instans dengan PuTTY dan Anda menerima pesan kesalahan “Server menutup koneksi jaringan secara tidak terduga”, verifikasi bahwa Anda telah mengaktifkan keepalive di halaman Koneksi Konfigurasi PuTTY agar koneksi tidak terputus. Beberapa server memutus koneksi klien saat tidak menerima data apa pun dalam periode waktu tertentu. Atur Detik untuk keepalive menjadi 59 detik.

Jika Anda masih mengalami masalah setelah mengaktifkan keepalive, coba nonaktifkan algoritma Nagle di halaman Koneksi Konfigurasi PuTTY.

# Kesalahan: Validasi kunci host gagal untuk EC2 Instance Connect

Jika Anda memutar kunci host instans, kunci host baru tidak secara otomatis diunggah ke database kunci host AWS tepercaya. Hal ini menyebabkan validasi kunci host gagal ketika Anda mencoba untuk terhubung ke instans menggunakan klien berbasis peramban EC2 Instance Connect, dan Anda tidak dapat terhubung ke instans Anda.

Untuk mengatasi kesalahan, Anda harus menjalankan skrip `eic_harvest_hostkeys` pada instans Anda, yang mengunggah kunci host baru Anda ke EC2 Instance Connect. Skrip ini terletak di `/opt/aws/bin/` di instans Amazon Linux 2, dan di `/usr/share/ec2-instance-connect/` pada instans Ubuntu.

Amazon Linux 2

Untuk mengatasi kesalahan kegagalan validasi kunci host pada instans Amazon Linux 2

1. Hubungkan ke instans Anda menggunakan SSH.

   Anda dapat terhubung dengan menggunakan EC2 Instance Connect CLI atau dengan menggunakan pasangan kunci SSH yang ditetapkan ke instans Anda saat Anda meluncurkannya dan nama pengguna default AMI yang Anda gunakan untuk meluncurkan instans. Untuk Amazon Linux 2, nama pengguna defaultnya adalah`ec2-user`.

   Misalnya, jika instans Anda diluncurkan menggunakan Amazon Linux 2, nama DNS publik instans Anda adalah `ec2-a-b-c-d.us-west-2.compute.amazonaws.com`, dan pasangan kuncinya adalah `my_ec2_private_key.pem`, gunakan perintah berikut untuk SSH ke dalam instans Anda:

   ```
   $ ssh -i my_ec2_private_key.pem ec2-user@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat Hubungkan ke instans Linux Anda menggunakan klien SSH.

2. Arahkan ke folder berikut.

   ```
   [ec2-user ~]$ cd /opt/aws/bin/
   ```

3. Jalankan perintah berikut di instans Anda.

```
[ec2-user ~]$ ./eic_harvest_hostkeys
```

Perhatikan bahwa panggilan yang berhasil tidak menghasilkan output.

Anda sekarang dapat menggunakan klien berbasis peramban EC2 Instance Connect untuk terhubung ke instans Anda.

Ubuntu

Untuk mengatasi kesalahan kegagalan validasi kunci host pada instans Ubuntu

1. Hubungkan ke instans Anda menggunakan SSH.

   Anda dapat terhubung dengan menggunakan EC2 Instance Connect CLI atau dengan menggunakan pasangan kunci SSH yang ditetapkan ke instans Anda saat Anda meluncurkannya dan nama pengguna default AMI yang Anda gunakan untuk meluncurkan instans. Untuk Ubuntu, nama pengguna defaultnya adalah`ubuntu`.

   Misalnya, jika instans Anda diluncurkan menggunakan Ubuntu, nama DNS publik instans Anda adalah `ec2-a-b-c-d.us-west-2.compute.amazonaws.com`, dan pasangan kuncinya adalah `my_ec2_private_key.pem`, gunakan perintah berikut untuk SSH ke dalam instans Anda:

   ```
   $ ssh -i my_ec2_private_key.pem ubuntu@ec2-a-b-c-d.us-
   west-2.compute.amazonaws.com
   ```

   Untuk informasi selengkapnya tentang menghubungkan ke instans Anda, lihat Hubungkan ke instans Linux Anda menggunakan klien SSH.

2. Arahkan ke folder berikut.

   ```
   [ec2-user ~]$ cd /usr/share/ec2-instance-connect/
   ```

3. Jalankan perintah berikut di instans Anda.

   ```
   [ec2-user ~]$ ./eic_harvest_hostkeys
   ```

   Perhatikan bahwa panggilan yang berhasil tidak menghasilkan output.

Anda sekarang dapat menggunakan klien berbasis peramban EC2 Instance Connect untuk terhubung ke instans Anda.

## Tidak dapat terhubung ke instans Ubuntu menggunakan EC2 Instance Connect

Jika Anda menggunakan EC2 Instance Connect untuk terhubung ke instans Ubuntu Anda dan mendapatkan kesalahan saat mencoba menghubungkan, Anda dapat menggunakan informasi berikut untuk mencoba memperbaiki masalah.

Kemungkinan penyebab

Paket `ec2-instance-connect` pada instans ini bukanlah versi terbaru.

Solusi

Perbarui paket `ec2-instance-connect` pada instans ke versi terbaru, sebagai berikut:

1. Hubungkan ke instans Anda menggunakan metode selain EC2 Instance Connect.
2. Jalankan perintah berikut pada instans Anda untuk memperbarui paket `ec2-instance-connect` ke versi terbaru.

```
apt update && apt upgrade
```

## Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance saya?

Jika Anda kehilangan kunci pribadi untuk sebuah EBS-backed instance, Anda dapat memperoleh kembali akses ke instans Anda. Anda harus menghentikan instans, mencopot volume root, dan melampirkannya ke instans lain sebagai volume data, ubah file `authorized_keys` dengan kunci publik baru, memindahkan volume kembali ke instans asli, lalu memulai ulang instans. Untuk informasi tentang peluncuran, penghubungan ke, dan penghentian instans selengkapnya, lihat Perubahan status instans Amazon EC2.

Prosedur ini hanya didukung untuk instans dengan volume root EBS. Jika instance memiliki volume root penyimpanan instans, Anda tidak dapat menggunakan prosedur ini untuk mendapatkan kembali

akses ke instans Anda; Anda harus memiliki kunci pribadi untuk terhubung ke instans. Untuk menentukan jenis volume root instans Anda, buka konsol Amazon EC2, pilih Instans, pilih instans, pilih tab Peny impanan, dan di bagian Detail perangkat Root, periksa nilai jenis perangkat Root.

Nilainya antara `EBS` atau.`INSTANCE-STORE`

Selain langkah-langkah berikut, ada cara lain untuk terhubung ke instans Linux Anda jika kehilangan kunci privat Anda. Untuk informasi selengkapnya, lihat Bagaimana cara saya terhubung ke instans Amazon EC2 jika saya kehilangan pasangan kunci SSH setelah peluncuran awal?

Langkah-langkah untuk menghubungkan ke EBS-backed instance dengan pasangan kunci yang berbeda

- Langkah 1: Buat pasangan kunci baru
- Langkah 2: Dapatkan informasi tentang instans asli dan volume root-nya
- Langkah 3: Hentikan instans asli
- Langkah 4: Luncurkan instans sementara
- Langkah 5: Copot volume root dari instans asli dan lampirkan ke instans sementara
- Langkah 6: Tambahkan kunci publik baru ke authorized_keys pada volume asli yang dipasang ke instance sementara
- Langkah 7: Lepaskan dan copot volume asli dari instans sementara, lalu lampirkan kembali ke instans asli
- Langkah 8: Hubungkan ke instans asli menggunakan pasangan kunci baru
- Langkah 9: Bersihkan

## Langkah 1: Buat pasangan kunci baru

Buat pasangan kunci baru menggunakan konsol Amazon EC2 atau alat pihak ketiga. Jika Anda ingin nama dari pasangan kunci baru Anda sama persis dengan kunci privat yang hilang, Anda harus menghapus pasangan kunci yang sudah ada terlebih dahulu. Untuk informasi tentang pembuatan pasangan kunci selengkapnya, lihat Membuat pasangan kunci menggunakan Amazon EC2 or Membuat pasangan kunci menggunakan alat pihak ketiga dan mengimpor kunci publik ke Amazon EC2.

## Langkah 2: Dapatkan informasi tentang instans asli dan volume root-nya

Catat informasi berikut karena Anda akan membutuhkannya untuk menyelesaikan prosedur ini.

Untuk mendapatkan informasi tentang instans asli Anda

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pilih Instans dalam panel navigasi, lalu pilih instans yang ingin Anda hubungkan. (Kami akan merujuknya sebagai instans asli.)
3. Pada tab Detail, catat ID instans dan ID AMI.
4. Pada tab Jaringan, catat Zona Ketersediaan.
5. Pada tab Penyimpanan, di bawah Nama perangkat root, catat nama perangkat untuk volume root (misalnya, `/dev/xvda`). Lalu, di bawah Perangkat blok, temukan nama perangkat ini dan catat ID volume (misalnya, vol-0a1234b5678c910de).

## Langkah 3: Hentikan instans asli

Pilih Status instans, Hentikan instans. Jika opsi ini dinonaktifkan, instans sudah dihentikan atau volume root adalah volume penyimpanan instans.

> **Warning**
>
> Saat Anda menghentikan instance, data pada volume penyimpanan instans hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

## Langkah 4: Luncurkan instans sementara

Untuk meluncurkan instans sementara

1. Di panel navigasi, pilih Instans, lalu pilih Luncurkan instans.
2. Di bagian Nama dan tanda, untuk Nama, masukkan Sementara.
3. Di bagian Gambar Aplikasi dan OS, pilih AMI yang sama dengan yang Anda gunakan untuk meluncurkan instans asli. Jika AMI ini tidak tersedia, Anda dapat membuat AMI yang dapat digunakan dari instans yang dihentikan. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.
4. Di bagian Tipe instans, pertahankan tipe instance default.
5. Di bagian Pasangan kunci, untuk Nama pasangan kunci, pilih pasangan kunci yang ada untuk digunakan atau buat yang baru.

6. Di bagian Pengaturan jaringan, pilih Edit, lalu untuk Subnet, pilih subnet di Zona Ketersediaan yang sama dengan instans asli.
7. Di panel Ringkasan, pilih Luncurkan.

### Langkah 5: Copot volume root dari instans asli dan lampirkan ke instans sementara

1. Di panel navigasi, pilih Volume dan pilih volume root untuk instance asli (Anda mencatat ID volumenya di langkah sebelumnya). Pilih Tindakan, Copot volume, lalu pilih Lepaskan. Tunggu status volume menjadi `available`. (Anda mungkin harus memilih ikon Segarkan.)
2. Dengan volume yang masih dipilih, pilih Tindakan, lalu pilih Lampirkan volume. Pilih ID instans dari instans sementara, catat nama perangkat yang ditentukan di bawah Nama perangkat (misalnya, `/dev/sdf`), lalu pilih Lampirkan volume.

> **Note**
>
> Jika Anda meluncurkan instans asli dari AWS Marketplace AMI dan volume berisi AWS Marketplace kode, Anda harus menghentikan instance sementara terlebih dahulu sebelum dapat melampirkan volume.

### Langkah 6: Tambahkan kunci publik baru ke **`authorized_keys`** pada volume asli yang dipasang ke instance sementara

1. Luncurkan ke instans sementara.
2. Dari instans sementara, pasang volume yang Anda lampirkan ke instans sehingga Anda dapat mengakses sistem file-nya. Misalnya, jika nama perangkat adalah `/dev/sdf`, gunakan perintah berikut untuk memasang volume sebagai `/mnt/tempvol`.

> **Note**
>
> Nama perangkat mungkin akan dimunculkan secara berbeda pada instans Anda. Misalnya, perangkat yang dipasang sebagai `/dev/sdf` dapat muncul sebagai `/dev/xvdf` pada instans. Beberapa versi Red Hat (atau variannya, seperti CentOS) bahkan dapat menambah huruf tambahan sebanyak 4 karakter, di mana `/dev/sdf` menjadi `/dev/xvdk`.

a. Gunakan perintah lsblk untuk menentukan apakah volume sudah dipartisi atau belum.

```
[ec2-user ~]$ lsblk
NAME    MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvda    202:0    0    8G  0 disk
##xvda1 202:1    0    8G  0 part /
xvdf    202:80   0  101G  0 disk
##xvdf1 202:81   0  101G  0 part
xvdg    202:96   0   30G  0 disk
```

Di contoh sebelumnya, `/dev/xvda` dan `/dev/xvdf` adalah volume yang sudah dipartisi, dan `/dev/xvdg` yang belum. Jika volume Anda dipartisi, Anda dapat memasang partisi tersebut (`/dev/xvdf1`) alih-alih perangkat mentah (`/dev/xvdf`) pada langkah berikutnya.

b. Buat direktori sementara untuk memasang volume.

```
[ec2-user ~]$ sudo mkdir /mnt/tempvol
```

c. Pasang volume (atau partisi) pada titik pasang sementara, menggunakan nama volume atau nama perangkat yang Anda identifikasi sebelumnya. Perintah yang diperlukan bergantung pada sistem file sistem operasi Anda. Perhatikan bahwa nama perangkat mungkin akan dimunculkan secara berbeda pada instans Anda. Lihat note di Langkah 6 untuk informasi selengkapnya.

- Amazon Linux, Ubuntu, dan Debian

```
[ec2-user ~]$ sudo mount /dev/xvdf1 /mnt/tempvol
```

- Amazon Linux 2, CentOS, SUSE Linux 12, dan RHEL 7.x

```
[ec2-user ~]$ sudo mount -o nouuid /dev/xvdf1 /mnt/tempvol
```

> **Note**
>
> Jika Anda mendapatkan kesalahan yang menyatakan bahwa sistem file rusak, jalankan perintah berikut untuk menggunakan utilitas fsck guna memeriksa sistem file dan memperbaiki masalah:

```
[ec2-user ~]$ sudo fsck /dev/xvdf1
```

3. Dari instans sementara, gunakan perintah berikut untuk memperbarui `authorized_keys` pada volume yang dipasang dengan kunci publik baru dari `authorized_keys` untuk instans sementara.

> **Important**
>
> Contoh berikut menggunakan nama pengguna Amazon Linux`ec2-user`. Anda mungkin perlu mengganti nama pengguna yang berbeda, seperti `ubuntu` untuk instance Ubuntu.

```
[ec2-user ~]$ cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/
authorized_keys
```

Jika penyalinan ini berhasil, Anda dapat melanjutkan ke langkah berikutnya.

(Opsional) Kecuali, jika Anda tidak memiliki izin untuk mengedit file di `/mnt/tempvol`, Anda harus memperbarui file menggunakan sudo, lalu memeriksa izin pada file untuk memverifikasi bahwa Anda dapat masuk ke instans asli. Gunakan perintah berikut untuk memeriksa izin pada file.

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
total 4
-rw------- 1 222 500 398 Sep 13 22:54 authorized_keys
```

Dalam contoh ini output, *222* adalah ID pengguna dan *500* merupakan ID grup. Berikutnya, gunakan sudo untuk menjalankan kembali perintah penyalinan yang gagal.

```
[ec2-user ~]$ sudo cp .ssh/authorized_keys /mnt/tempvol/home/ec2-user/.ssh/
authorized_keys
```

Jalankan lagi akun perintah berikut untuk menentukan apakah izin sudah berubah atau belum.

```
[ec2-user ~]$ sudo ls -l /mnt/tempvol/home/ec2-user/.ssh
```

Jika ID pengguna dan ID grup telah berubah, gunakan perintah berikut untuk memulihkannya.

```
[ec2-user ~]$ sudo chown 222:500 /mnt/tempvol/home/ec2-user/.ssh/authorized_keys
```

## Langkah 7: Lepaskan dan copot volume asli dari instans sementara, lalu lampirkan kembali ke instans asli

1. Dari instans sementara, lepas volume yang Anda lampirkan ke instans sehingga Anda dapat melampirkannya kembali ke instans asli. Misalnya, gunakan perintah berikut untuk melepaskan volume pada `/mnt/tempvol`.

   ```
   [ec2-user ~]$ sudo umount /mnt/tempvol
   ```

2. Lepaskan volume dari instance sementara (Anda melepasnya pada langkah sebelumnya): Dari konsol Amazon EC2, pilih Volume di panel navigasi, pilih volume root untuk instance asli (Anda mencatat ID volume pada langkah sebelumnya), pilih Tindakan, Lepaskan volume , lalu pilih Lepaskan. Tunggu status volume menjadi `available`. (Anda mungkin harus memilih ikon Segarkan.)
3. Lampirkan kembali volume ke instans asli: Dengan volume yang masih dipilih, pilih Tindakan, Lampirkan Volume. Pilih ID instance instans asli, tentukan nama perangkat yang Anda catat sebelumnya di Langkah 2 untuk lampiran volume root asli (`/dev/sda1`atau`/dev/xvda`), lalu pilih Lamp irkan volume.

   > **Important**
   >
   > Jika Anda tidak menentukan nama perangkat yang sama dengan lampiran asli, Anda tidak dapat memulai instans asli. Amazon EC2 mengharapkan volume root pada `sda1` atau`/dev/xvda`.

## Langkah 8: Hubungkan ke instans asli menggunakan pasangan kunci baru

Pilih instans asli, pilih Status instans, Mulai instans. Setelah instans memasuki status `running`, Anda dapat terhubung menggunakan file kunci privat untuk pasangan kunci baru Anda.

> **Note**
>
> Jika nama pasangan kunci baru Anda dan file kunci privat yang terkait berbeda dari nama pasangan kunci asli, pastikan Anda menentukan nama file kunci privat baru saat terhubung ke instans Anda.

### Langkah 9: Bersihkan

(Opsional) Anda dapat mengakhiri instans sementara jika tidak menggunakannya lagi. Pilih instance sementara, dan pilih Status  instans, H  entikan (hapus) instance.

## Memecahkan masalah instans Amazon EC2 Linux dengan pemeriksaan status yang gagal

Informasi berikut dapat membantu Anda memecahkan masalah jika instance Linux Anda gagal dalam pemeriksaan status. Pertama-tama, tentukan apakah aplikasi Anda menunjukkan adanya masalah. Jika Anda memverifikasi bahwa instans tidak menjalankan aplikasi Anda seperti yang diharapkan, tinjau informasi pemeriksaan status dan log sistem.

Untuk contoh masalah yang dapat menyebabkan pemeriksaan status gagal, lihat Pemeriksaan status untuk instans Amazon EC2.

Daftar Isi


- Meninjau informasi pemeriksaan status
- Mengambil log sistem
- Memecahkan masalah kesalahan log sistem untuk instans Linux
- Kehabisan memori: hentikan proses
- KESALAHAN: mmu_update gagal (Pembaruan manajemen memori gagal)
- I/O kesalahan (kegagalan perangkat blok)
- I/O KESALAHAN: bukan disk lokal maupun jarak jauh (Perangkat blok terdistribusi rusak)
- request_module: modprobe loop runaway (Melakukan loop modprobe kernel warisan pada versi Linux yang lebih lawas)
- “FATAL: kernel terlalu lawas” dan “fsck: Tidak ada file atau direktori tersebut saat mencoba membuka /dev” (Kernel dan AMI tidak cocok)

- “FATAL: Tidak dapat memuat/lib/modules" atau BusyBox "" (Modul kernel hilang)
- KESALAHAN Kernel tidak valid (kernel tidak kompatibel dengan EC2)
- fsck: Tidak ada file atau direktori tersebut saat mencoba membuka... (Sistem file tidak ditemukan)
- Kesalahan umum saat memasang sistem file (kegagalan pemasangan)
- VFS: Tidak dapat memasang fs root pada blok yang tidak diketahui (Sistem file root tidak cocok)
- Kesalahan: Tidak dapat menentukan major/minor jumlah perangkat root... (Ketidak system/device cocokan file root)
- XENBUS: Perangkat tanpa driver...
- ... hari tanpa diperiksa, pemeriksaan paksa (Diperlukan pemeriksaan sistem file)
- fsck mati dengan status keluar... (Perangkat tidak ada)
- Prompt GRUB (gru > bdom)
- Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. (Alamat Hard-coded MAC)
- Tidak dapat memuat Kebijakan SELinux. Mesin berada dalam mode pemberlakuan. Menghentikan sekarang. (Kesalahan konfigurasi SELinux)
- XENBUS: Waktu menghubungkan ke perangkat habis (waktu habis Xenbus)

## Meninjau informasi pemeriksaan status

Untuk menyelidiki instans yang terganggu menggunakan konsol Amazon EC2

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Pada panel navigasi, pilih Instans, kemudian pilih instans Anda.
3. Pilih tab Status dan alarm untuk melihat hasil individual untuk semua pemeriksaan status Sistem, Pemeriksaan status Inst ans , dan Pemeriksaan status EBS Terlampir.

Jika pemeriksaan status gagal, Anda dapat mencoba salah satu opsi berikut:

- Buat alarm untuk memulihkan instance sebagai respons terhadap pemeriksaan status yang gagal. Untuk informasi selengkapnya, lihat Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans.
- (Pemeriksaan status instans) Jika Anda mengubah jenis inst Nitro-based ans menjadi instance, pemeriksaan status gagal jika Anda bermigrasi dari instans yang tidak memiliki driver ENA dan

NVMe yang diperlukan. Untuk informasi selengkapnya, lihat Kompatibilitas untuk mengubah tipe instans.

- Untuk contoh dengan volume root EBS, hentikan dan mulai ulang instance. Untuk informasi selengkapnya, lihat Menghentikan dan memulai instans Amazon EC2.
- Untuk instance dengan volume root penyimpanan instans, hentikan instance dan luncurkan instance pengganti. Untuk informasi selengkapnya, lihat Menghentikan instans Amazon EC2.
- Tunggu hingga Amazon EC2 menyelesaikan masalah.
- Hubun Dukungan gi atau posting masalah Anda ke AWS re: post.
- Jika instance Anda berada dalam grup Penskalaan Otomatis:
  - (Pemeriksaan status sistem dan pemeriksaan status instans) Secara default, Amazon EC2 Auto Scaling secara otomatis meluncurkan instans pengganti. Untuk informasi selengkapnya, lihat Pemeriksaan kesehatan untuk instans dalam grup Penskalaan Otom atis di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
  - (Pemeriksaan status EBS terlampir) Anda harus mengonfigurasi Amazon EC2 Auto Scaling untuk meluncurkan instans pengganti secara otomatis. Untuk informasi selengkapnya, lihat Mem antau dan mengganti instans Auto Scaling dengan volume Amazon EBS yang rusak di Panduan Pengguna Penskalaan Otomatis Amazon EC2.
- Mengambil log sistem dan mencari kesalahan. Untuk informasi selengkapnya, lihat Mengambil log sistem.

## Mengambil log sistem

Jika pemeriksaan status instans gagal, Anda dapat melakukan boot ulang instans dan mengambil log sistem. Log tersebut mungkin memperlihatkan kesalahan yang dapat membantu Anda memecahkan masalah. Boot ulang akan menghapus informasi yang tidak diperlukan dari log.

Untuk melakukan boot ulang instans dan mengambil log sistem

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans, dan pilih instans Anda.
3. Pilih Status instans, Boot ulang instans. Mungkin diperlukan waktu beberapa menit untuk melakukan boot ulang instans Anda.
4. Verifikasi bahwa masalah masih ada. Pada beberapa kasus, boot ulang dapat menyelesaikan masalah.

5. Ketika instans berada dalam status `running`, pilih Tindakan, Memantau dan memecahkan masalah, Dapatkan log sistem.
6. Tinjau log yang muncul di layar, dan gunakan daftar pernyataan kesalahan log sistem yang diketahui di bawah ini untuk memecahkan masalah Anda.
7. Jika masalah belum teratasi, Anda dapat membuat pos masalah Anda ke AWS re:Post.

## Memecahkan masalah kesalahan log sistem untuk instans Linux

Untuk instans Linux yang gagal dalam pemeriksaan status instance, seperti pemeriksaan jangkauan instans, verifikasi bahwa Anda mengikuti langkah-langkah di atas untuk mengambil log sistem. Daftar berikut berisi beberapa kesalahan log sistem umum dan tindakan yang disarankan yang dapat Anda ambil guna mengatasi masalah pada setiap kesalahan.

Kesalahan Memori

- Kehabisan memori: hentikan proses
- KESALAHAN: mmu_update gagal (Pembaruan manajemen memori gagal)

Kesalahan Perangkat

- I/O kesalahan (kegagalan perangkat blok)
- I/O KESALAHAN: bukan disk lokal maupun jarak jauh (Perangkat blok terdistribusi rusak)

Kesalahan Kernel

- request_module: modprobe loop runaway (Melakukan loop modprobe kernel warisan pada versi Linux yang lebih lawas)
- “FATAL: kernel terlalu lawas” dan “fsck: Tidak ada file atau direktori tersebut saat mencoba membuka /dev” (Kernel dan AMI tidak cocok)
- “FATAL: Tidak dapat memuat/lib/modules" atau BusyBox "" (Modul kernel hilang)
- KESALAHAN Kernel tidak valid (kernel tidak kompatibel dengan EC2)

Kesalahan Sistem File

- fsck: Tidak ada file atau direktori tersebut saat mencoba membuka... (Sistem file tidak ditemukan)

- Kesalahan umum saat memasang sistem file (kegagalan pemasangan)
- VFS: Tidak dapat memasang fs root pada blok yang tidak diketahui (Sistem file root tidak cocok)
- Kesalahan: Tidak dapat menentukan major/minor jumlah perangkat root... (Ketidak system/device cocokan file root)
- XENBUS: Perangkat tanpa driver...
- ... hari tanpa diperiksa, pemeriksaan paksa (Diperlukan pemeriksaan sistem file)
- fsck mati dengan status keluar... (Perangkat tidak ada)

Kesalahan Sistem Operasi

- Prompt GRUB (gru > bdom)
- Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. (Alamat Hard-coded MAC)
- Tidak dapat memuat Kebijakan SELinux. Mesin berada dalam mode pemberlakuan. Menghentikan sekarang. (Kesalahan konfigurasi SELinux)
- XENBUS: Waktu menghubungkan ke perangkat habis (waktu habis Xenbus)

## Kehabisan memori: hentikan proses

Kesalahan kehabisan memori diindikasikan oleh entri log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
[115879.769795] Out of memory: kill process 20273 (httpd) score 1285879
or a child
[115879.769795] Killed process 1917 (php-cgi) vsz:467184kB, anon-
rss:101196kB, file-rss:204kB
```

### Potensi penyebab

Memori habis

### Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Lakukan salah satu tindakan berikut: |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | • Hentikan instans, dan modifikasi instans untuk menggunakan tipe instans yang berbeda, dan mulai lagi instans tersebut. Misalnya, tipe instans yang lebih besar atau instans memori yang dioptimalkan.<br>• Boot ulang instans untuk mengembalikannya ke status tidak terganggu. Masalah mungkin akan terjadi lagi kecuali Anda mengubah tipe instans. |
| Didukung penyimpanan instans | Lakukan salah satu dari berikut ini:<br>• Akhiri instans dan luncurkan instans baru, dengan menentukan tipe instans yang berbeda. Misalnya, tipe instans yang lebih besar atau instans memori yang dioptimal kan.<br>• Boot ulang instans untuk mengembalikannya ke status tidak terganggu. Masalah mungkin akan terjadi lagi kecuali Anda mengubah tipe instans. |

## KESALAHAN: mmu_update gagal (Pembaruan manajemen memori gagal)

Kegagalan pembaruan manajemen memori diindikasikan oleh entri log sistem yang serupa dengan yang berikut ini:

```
...
Press `ESC' to enter the menu... 0   [H[J  Booting 'Amazon Linux 2011.09
 (2.6.35.14-95.38.amzn1.i686)'


root (hd0)

 Filesystem type is ext2fs, using whole disk
```

```
kernel /boot/vmlinuz-2.6.35.14-95.38.amzn1.i686 root=LABEL=/ console=hvc0 LANG=

en_US.UTF-8 KEYTABLE=us

initrd /boot/initramfs-2.6.35.14-95.38.amzn1.i686.img

ERROR: mmu_update failed with rc=-22
```

### Potensi penyebab

Masalah dengan Amazon Linux

### Tindakan yang disarankan

Posting masalah Anda ke AWS re:post atau kontak. Dukungan

## I/O kesalahan (kegagalan perangkat blok)

Kes input/output alahan ditunjukkan oleh entri log sistem yang mirip dengan contoh berikut:

```
[9943662.053217] end_request: I/O error, dev sde, sector 52428288
[9943664.191262] end_request: I/O error, dev sde, sector 52428168
[9943664.191285] Buffer I/O error on device md0, logical block 209713024
[9943664.191297] Buffer I/O error on device md0, logical block 209713025
[9943664.191304] Buffer I/O error on device md0, logical block 209713026
[9943664.191310] Buffer I/O error on device md0, logical block 209713027
[9943664.191317] Buffer I/O error on device md0, logical block 209713028
[9943664.191324] Buffer I/O error on device md0, logical block 209713029
[9943664.191332] Buffer I/O error on device md0, logical block 209713030
[9943664.191339] Buffer I/O error on device md0, logical block 209713031
[9943664.191581] end_request: I/O error, dev sde, sector 52428280
[9943664.191590] Buffer I/O error on device md0, logical block 209713136
[9943664.191597] Buffer I/O error on device md0, logical block 209713137
[9943664.191767] end_request: I/O error, dev sde, sector 52428288
[9943664.191970] end_request: I/O error, dev sde, sector 52428288
[9943664.192143] end_request: I/O error, dev sde, sector 52428288
[9943664.192949] end_request: I/O error, dev sde, sector 52428288
[9943664.193112] end_request: I/O error, dev sde, sector 52428288
[9943664.193266] end_request: I/O error, dev sde, sector 52428288
...
```

## Potensi penyebab

| Jenis instans | Potensi penyebab |
| --- | --- |
| Amazon EBS-backed | Volume Amazon EBS yang gagal |
| Didukung penyimpanan instans | Drive fisik yang gagal |

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut:<br>1. Hentikan instans.<br>2. Lepaskan lampiran volume.<br>3. Coba pulihkan volume.<br>**Note**<br>Disarankan untuk sering membuat snapshot volume Amazon EBS. Hal ini secara signifikan mengurangi risiko kehilangan data sebagai akibat dari kegagalan.<br>4. Re-attach volume untuk instance.<br>5. Mulai instans. |
| Didukung penyimpanan instans | Akhiri instans dan luncurkan instans baru.<br>**Note**<br>Data tidak dapat dipulihkan. Pulihkan dari cadangan. |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | **Note**<br>Disarankan agar menggunakan Amazon S3 atau Amazon EBS untuk pencadangan. Volume penyimpanan instans secara langsung terikat dengan kegagalan host tunggal dan disk tunggal. |

## I/O KESALAHAN: bukan disk lokal maupun jarak jauh (Perangkat blok terdistribusi rusak)

Kes input/output alahan pada perangkat ditunjukkan oleh entri log sistem yang mirip dengan contoh berikut:

```
...
block drbd1: Local IO failed in request_timer_fn. Detaching...

Aborting journal on device drbd1-8.

block drbd1: IO ERROR: neither local nor remote disk

Buffer I/O error on device drbd1, logical block 557056

lost page write due to I/O error on drbd1

JBD2: I/O error detected when updating journal superblock for drbd1-8.
```

### Potensi penyebab

| Jenis instans | Potensi penyebab |
| --- | --- |
| Amazon EBS-backed | Volume Amazon EBS yang gagal |
| Didukung penyimpanan instans | Drive fisik yang gagal |

### Tindakan yang disarankan

Akhiri instans dan luncurkan instans baru.

Untuk EBS-backed contoh Amazon, Anda dapat memulihkan data dari snapshot terbaru dengan membuat gambar darinya. Data apa pun yang ditambahkan setelah snapshot tidak dapat dipulihkan.

## request_module: modprobe loop runaway (Melakukan loop modprobe kernel warisan pada versi Linux yang lebih lawas)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini. Penggunaan kernel Linux yang tidak stabil atau lawas (misalnya 2.6.16-xenU) dapat menyebabkan kondisi loop yang tidak dapat dihentikan saat dimulai.

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007

BIOS-provided physical RAM map:

 Xen: 0000000000000000 - 0000000026700000 (usable)

0MB HIGHMEM available.
...

request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c

request_module: runaway loop modprobe binfmt-464c
```

### Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan kernel yang lebih baru, baik GRUB-based atau statis, menggunakan salah satu opsi berikut: |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | Opsi 1: Akhiri instans dan luncurkan instans baru, dengan menentukan parameter `-kernel` dan `-ramdisk`.<br><br>Opsi 2:<br><br>1. Hentikan instans.<br>2. Modifikasi atribut kernel dan ramdisk untuk menggunakan kernel yang lebih baru.<br>3. Mulai instans. |
| Didukung penyimpanan instans | Akhiri instans dan luncurkan instans baru, dengan menentukan parameter `-kernel` dan `-ramdisk`. |

## “FATAL: kernel terlalu lawas” dan “fsck: Tidak ada file atau direktori tersebut saat mencoba membuka /dev” (Kernel dan AMI tidak cocok)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
Linux version 2.6.16.33-xenU (root@dom0-0-50-45-1-a4-ee.z-2.aes0.internal)
(gcc version 4.1.1 20070105 (Red Hat 4.1.1-52)) #2 SMP Wed Aug 15 17:27:36 SAST 2007
...
FATAL: kernel too old
Kernel panic - not syncing: Attempted to kill init!
```

### Potensi penyebab

Kernel dan userland tidak kompatibel

### Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut: |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | 1. Hentikan instans.<br>2. Modifikasi konfigurasi untuk menggunakan kernel yang lebih baru.<br>3. Mulai instans. |
| Didukung penyimpanan instans | Gunakan prosedur berikut:<br>1. Buat AMI yang menggunakan kernel yang lebih baru.<br>2. Akhiri instans.<br>3. Mulai instans baru dari AMI yang Anda buat. |

## “FATAL: Tidak dapat memuat/lib/modules" atau BusyBox "" (Modul kernel hilang)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
[    0.370415] Freeing unused kernel memory: 1716k freed
Loading, please wait...
WARNING: Couldn't open directory /lib/modules/2.6.34-4-virtual: No such file or
 directory
FATAL: Could not open /lib/modules/2.6.34-4-virtual/modules.dep.temp for writing: No
 such file or directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
Couldn't get a file descriptor referring to the console
Begin: Loading essential drivers... ...
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
Done.
Begin: Running /scripts/init-premount ...
Done.
Begin: Mounting root file system... ...
Begin: Running /scripts/local-top ...
Done.
```

```
Begin: Waiting for root file system... ...
Done.
Gave up waiting for root device.  Common problems:
 - Boot args (cat /proc/cmdline)
   - Check rootdelay= (did the system wait long enough?)
   - Check root= (did the system wait for the right device?)
 - Missing modules (cat /proc/modules; ls /dev)
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
FATAL: Could not load /lib/modules/2.6.34-4-virtual/modules.dep: No such file or
 directory
ALERT! /dev/sda1 does not exist. Dropping to a shell!


BusyBox v1.13.3 (Ubuntu 1:1.13.3-1ubuntu5) built-in shell (ash)
Enter 'help' for a list of built-in commands.

(initramfs)
```

## Potensi penyebab

Satu atau lebih kondisi berikut dapat menyebabkan masalah ini:

- Ramdisk tidak ada
- Modul yang benar dari ramdisk tidak ada
- Volume root Amazon EBS tidak dilampirkan dengan benar sebagai `/dev/sda1`

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut:<br>1. Pilih ramdisk yang sudah benar untuk volume Amazon EBS.<br>2. Hentikan instans.<br>3. Lepaskan lampiran volume dan perbaiki.<br>4. Lampirkan volume ke instans.<br>5. Mulai instans. |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | 6. Modifikasi AMI untuk menggunakan ramdisk yang telah diperbaiki. |
| Didukung penyimpanan instans | Gunakan prosedur berikut:<br>1. Akhiri instans dan luncurkan instans baru dengan ramdisk yang benar.<br>2. Buat AMI baru dengan ramdisk yang benar. |

## KESALAHAN Kernel tidak valid (kernel tidak kompatibel dengan EC2)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
...
root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz root=/dev/sda1 ro

initrd /initrd.img

ERROR Invalid kernel: elf_xen_note_check: ERROR: Will only load images
built for the generic loader or Linux images
xc_dom_parse_image returned -1

Error 9: Unknown boot failure

  Booting 'Fallback'

root (hd0)

 Filesystem type is ext2fs, using whole disk

kernel /vmlinuz.old root=/dev/sda1 ro

Error 15: File not found
```

### Potensi penyebab

Salah satu atau kedua kondisi berikut dapat menyebabkan masalah ini:

- Kernel yang disediakan tidak didukung oleh GRUB
- Kernel pengganti tidak ada

### Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
|---|---|
| Amazon EBS-backed | Gunakan prosedur berikut:<br>1. Hentikan instans.<br>2. Ganti dengan kernel yang berfungsi.<br>3. Instal kernel pengganti.<br>4. Modifikasi AMI dengan memperbaiki kernel. |
| Didukung penyimpanan instans | Gunakan prosedur berikut:<br>1. Akhiri instans dan luncurkan instans baru dengan kernel yang benar.<br>2. Buat AMI baru dengan kernel yang benar.<br>3. (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan Dukungan. |

## fsck: Tidak ada file atau direktori tersebut saat mencoba membuka... (Sistem file tidak ditemukan)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
  Welcome to Fedora
  Press 'I' to enter interactive startup.
Setting clock : Wed Oct 26 05:52:05 EDT 2011 [  OK  ]

Starting udev: [  OK  ]
```

```
Setting hostname localhost:  [  OK  ]

No devices found
Setting up Logical Volume Management: File descriptor 7 left open
  No volume groups found
[  OK  ]

Checking filesystems
Checking all file systems.
[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1
/dev/sda1: clean, 82081/1310720 files, 2141116/2621440 blocks
[/sbin/fsck.ext3 (1) -- /mnt/dbbackups] fsck.ext3 -a /dev/sdh
fsck.ext3: No such file or directory while trying to open /dev/sdh

/dev/sdh:
The superblock could not be read or does not describe a correct ext2
filesystem.  If the device is valid and it really contains an ext2
filesystem (and not swap or ufs or something else), then the superblock
is corrupt, and you might try running e2fsck with an alternate superblock:
    e2fsck -b 8193 <device>

[FAILED]


*** An error occurred during the file system check.
*** Dropping you to a shell; the system will reboot
*** when you leave the shell.
Give root password for maintenance
(or type Control-D to continue):
```

## Potensi penyebab

- Ada bug dalam definisi sistem file ramdisk/etc/fstab
- Definisi sistem file yang salah konfigurasi di/etc/fstab
- Missing/failed mengemudi

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut: |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | 1. Hentikan instance, lepaskan volume root, repair/modify /etc/fstab volume, pasang volume ke instance, dan mulai instance.<br>2. Perbaiki ramdisk untuk menyertakan/yang dimodifikasi etc/fstab (jika ada).<br>3. Modifikasi AMI untuk menggunakan ramdisk yang lebih baru.<br><br>Bidang keenam di fstab menentukan persyaratan ketersediaan pemasangan –nilai bukan nol mengimplikasikan bahwa fsck akan dilakukan pada volume tersebut dan harus berhasil. Penggunaan bidang ini dapat menjadi masalah di Amazon EC2 karena kegagalan biasanya mengakibatkan perintah konsol interaktif yang saat ini tidak tersedia di Amazon EC2. Berhati-hatilah dengan fitur ini dan baca halaman manual Linux untuk fstab. |
| Didukung penyimpanan instans | Gunakan prosedur berikut:<br><br>1. Akhiri instans dan luncurkan instans baru.<br>2. Lepaskan lampiran volume Amazon EBS yang tidak sesuai dan boot ulang instans.<br>3. (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan Dukungan. |

## Kesalahan umum saat memasang sistem file (kegagalan pemasangan)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
Loading xenblk.ko module
xen-vbd: registered block device major 8
```

```
Loading ehci-hcd.ko module
Loading ohci-hcd.ko module
Loading uhci-hcd.ko module
USB Universal Host Controller Interface driver v3.0

Loading mbcache.ko module
Loading jbd.ko module
Loading ext3.ko module
Creating root device.
Mounting root filesystem.
kjournald starting.  Commit interval 5 seconds

EXT3-fs: mounted filesystem with ordered data mode.

Setting up other filesystems.
Setting up new root fs
no fstab.sys, mounting internal defaults
Switching to new root and running init.
unmounting old /dev
unmounting old /proc
unmounting old /sys
mountall:/proc: unable to mount: Device or resource busy
mountall:/proc/self/mountinfo: No such file or directory
mountall: root filesystem isn't mounted
init: mountall main process (221) terminated with status 1

General error mounting filesystems.
A maintenance shell will now be started.
CONTROL-D will terminate this shell and re-try.
Press enter for maintenance
(or type Control-D to continue):
```

## Potensi penyebab

| Jenis instans | Potensi penyebab |
| --- | --- |
| Amazon EBS-backed | • Volume Amazon EBS yang terlepas lampirannya atau gagal.<br>• Sistem file rusak. |

| Jenis instans | Potensi penyebab |
| --- | --- |
| | • Kombinasi ramdisk dan AMI yang tidak cocok (seperti ramdisk Debian dengan SUSE AMI). |
| Didukung penyimpanan instans | • Drive yang gagal.<br>• Sistem file yang rusak.<br>• Ramdisk dan kombinasi yang tidak cocok (misalnya ramdisk Debian dengan SUSE AMI). |

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut:<br>1. Hentikan instans.<br>2. Lepaskan lampiran volume root.<br>3. Lampirkan volume root ke instans yang berfungsi yang diketahui.<br>4. Jalankan pemeriksaan sistem file (fsck -a /dev/...).<br>5. Perbaiki kesalahan apa pun.<br>6. Lepaskan lampiran volume dari instans yang berfungsi yang diketahui.<br>7. Lampirkan volume ke instans yang dihentikan.<br>8. Mulai instans.<br>9. Periksa ulang status instans. |
| Didukung penyimpanan instans | Cobalah salah satu cara berikut ini:<br>• Mulai instans baru. |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | • (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan Dukungan. |

## VFS: Tidak dapat memasang fs root pada blok yang tidak diketahui (Sistem file root tidak cocok)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
 20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007
...
Kernel command line:  root=/dev/sda1 ro 4
...
Registering block device major 8
...
Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(8,1)
```

### Potensi penyebab

| Jenis instans | Potensi penyebab |
| --- | --- |
| Amazon EBS-backed | • Perangkat tidak terlampir dengan benar.<br>• Perangkat root tidak terpasang pada titik perangkat yang benar.<br>• Sistem file tidak dalam format yang diharapkan.<br>• Penggunaan kernel warisan (seperti 2.6.16-XenU).<br>• Pembaruan kernel terbaru di instans Anda (pembaruan yang salah, atau bug pembaruan) |
| Didukung penyimpanan instans | Kegagalan perangkat keras. |

### Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Lakukan salah satu tindakan berikut:<br>• Hentikan lalu mulai ulang instans.<br>• Ubah volume root untuk melampirkan pada titik perangkat yang benar, mungkin/al dev/sda1 ih-alih/dev/sda.<br>• Hentikan dan modifikasi untuk menggunakan kernel modern.<br>• Lihat dokumentasi untuk distribusi Linux Anda guna memeriksa bug pembaruan yang diketahui. Ubah atau instal ulang kernel. |
| Didukung penyimpanan instans | Akhiri instans dan luncurkan instans baru dengan kernel modern. |

## Kesalahan: Tidak dapat menentukan major/minor jumlah perangkat root... (Ketidak system/device cocokan file root)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
...
XENBUS: Device with no driver: device/vif/0
XENBUS: Device with no driver: device/vbd/2048
drivers/rtc/hctosys.c: unable to open rtc device (rtc0)
Initializing network drop monitor service
Freeing unused kernel memory: 508k freed
:: Starting udevd...
done.
:: Running Hook [udev]
:: Triggering uevents...<30>udevd[65]: starting version 173
done.
Waiting 10 seconds for device /dev/xvda1 ...
Root device '/dev/xvda1' doesn't exist. Attempting to create it.
ERROR: Unable to determine major/minor number of root device '/dev/xvda1'.
```

```
You are being dropped to a recovery shell
    Type 'exit' to try and continue booting
sh: can't access tty; job control turned off
[ramfs /]#
```

### Potensi penyebab

- Driver perangkat blok virtual tidak ada atau salah konfigurasi
- Konflik enumerasi perangkat (sda versus xvda atau sda alih-alih sda1)
- Pilihan yang salah untuk kernel instans

### Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut:<br>1. Hentikan instans.<br>2. Lepaskan lampiran volume.<br>3. Perbaiki masalah pemetaan perangkat.<br>4. Mulai instans.<br>5. Modifikasi AMI untuk menangani masalah pemetaan perangkat. |
| Didukung penyimpanan instans | Gunakan prosedur berikut:<br>1. Buat AMI baru dengan perbaikan yang sesuai (petakan perangkat blok dengan benar).<br>2. Akhiri instans dan luncurkan instans baru dari AMI yang Anda buat. |

## XENBUS: Perangkat tanpa driver...

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
XENBUS: Device with no driver: device/vbd/2048
drivers/rtc/hctosys.c: unable to open rtc device (rtc0)
Initializing network drop monitor service
Freeing unused kernel memory: 508k freed
:: Starting udevd...
done.
:: Running Hook [udev]
:: Triggering uevents...<30>udevd[65]: starting version 173
done.
Waiting 10 seconds for device /dev/xvda1 ...
Root device '/dev/xvda1' doesn't exist. Attempting to create it.
ERROR: Unable to determine major/minor number of root device '/dev/xvda1'.
You are being dropped to a recovery shell
    Type 'exit' to try and continue booting
sh: can't access tty; job control turned off
[ramfs /]#
```

## Potensi penyebab

- Driver perangkat blok virtual tidak ada atau salah konfigurasi
- Konflik enumerasi perangkat (sda versus xvda)
- Pilihan yang salah untuk kernel instans

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut:<br>1. Hentikan instans.<br>2. Lepaskan lampiran volume.<br>3. Perbaiki masalah pemetaan perangkat.<br>4. Mulai instans.<br>5. Modifikasi AMI untuk menangani masalah pemetaan perangkat. |
| Didukung penyimpanan instans | Gunakan prosedur berikut: |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | 1. Buat AMI dengan perbaikan yang sesuai (petakan perangkat blok dengan benar).<br>2. Akhiri instans dan luncurkan instans baru menggunakan AMI yang telah Anda buat. |

## ... hari tanpa diperiksa, pemeriksaan paksa (Diperlukan pemeriksaan sistem file)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
...
Checking filesystems
Checking all file systems.
[/sbin/fsck.ext3 (1) -- /] fsck.ext3 -a /dev/sda1
/dev/sda1 has gone 361 days without being checked, check forced
```

### Potensi penyebab

Waktu pemeriksaan sistem file telah berlalu; pemeriksaan paksa sistem file sedang dilakukan.

### Tindakan yang disarankan

- Tunggu hingga pemeriksaan sistem file selesai. Pemeriksaan sistem file dapat memakan waktu yang lama bergantung pada ukuran sistem file root.
- Modifikasi sistem file Anda untuk menghapus pemberlakuan pemeriksaan sistem file (fsck) menggunakan tune2fs atau alat yang sesuai untuk sistem file Anda.

## fsck mati dengan status keluar... (Perangkat tidak ada)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
Cleaning up ifupdown....
Loading kernel modules...done.
...
Activating lvm and md swap...done.
Checking file systems...fsck from util-linux-ng 2.16.2
```

```
/sbin/fsck.xfs: /dev/sdh does not exist
fsck died with exit status 8
[31mfailed (code 8).[39;49m
```

## Potensi penyebab

- Ramdisk mencari drive yang tidak ada
- Pemeriksaan paksa konsistensi sistem file dilakukan
- Drive gagal atau terlepas lampirannya

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Coba salah satu atau beberapa cara berikut untuk menyelesaikan masalah:<br>• Hentikan instans, lampirkan volume ke instans yang sudah ada yang sedang berjalan.<br>• Jalankan pemeriksaan konsistensi secara manual.<br>• Perbaiki ramdisk untuk menyertakan utilitas yang relevan.<br>• Modifikasi parameter penyetelan sistem file untuk menghapus persyaratan konsistensi (tidak disarankan). |
| Didukung penyimpanan instans | Coba salah satu atau beberapa cara berikut untuk menyelesaikan masalah:<br>• Buat ulang paket ramdisk dengan alat yang benar.<br>• Modifikasi parameter penyetelan sistem file untuk menghapus persyaratan konsistensi (tidak disarankan). |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | • Akhiri instans dan luncurkan instans baru.<br>• (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan Dukungan. |

## Prompt GRUB (gru > bdom)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
    GNU GRUB  version 0.97  (629760K lower / 0K upper memory)

       [ Minimal BASH-like line editing is supported.   For

         the   first   word,  TAB  lists  possible  command

         completions.  Anywhere else TAB lists the possible

         completions of a device/filename. ]

grubdom>
```

### Potensi penyebab

| Jenis instans | Potensi penyebab |
| --- | --- |
| Amazon EBS-backed | • File konfigurasi GRUB tidak ada.<br>• Citra GRUB yang digunakan salah, mengharapkan file konfigurasi GRUB berada di lokasi yang berbeda.<br>• Sistem file yang tidak didukung digunakan untuk menyimpan file konfigurasi GRUB Anda (misalnya mengonversi sistem file root ke tipe yang tidak didukung oleh versi GRUB sebelumnya). |
| Didukung penyimpanan instans | • File konfigurasi GRUB tidak ada. |

| Jenis instans | Potensi penyebab |
| --- | --- |
| | • Citra GRUB yang digunakan salah, mengharapkan file konfigurasi GRUB berada di lokasi yang berbeda.<br>• Sistem file yang tidak didukung digunakan untuk menyimpan file konfigurasi GRUB Anda (misalnya mengonversi sistem file root ke tipe yang tidak didukung oleh versi GRUB sebelumnya). |

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Opsi 1: Modifikasi AMI dan luncurkan ulang instans:<br>1. Ubah AMI sumber untuk membuat file konfigurasi GRUB di lokasi standar (/boot/grub/menu.lst).<br>2. Verifikasi bahwa versi GRUB Anda mendukung tipe sistem file yang mendasarinya dan tingkatkan GRUB jika perlu.<br>3. Pilih citra GRUB yang sesuai, (drive hd0-1st atau hd00 – drive pertama, partisi pertama).<br>4. Akhiri instans dan luncurkan instans baru menggunakan AMI yang Anda buat.<br><br>Opsi 2: Perbaiki instans yang sudah ada:<br>1. Hentikan instans.<br>2. Lepaskan lampiran sistem file root.<br>3. Lampirkan sistem file root ke instans yang berfungsi yang diketahui. |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | 4. Pasang sistem file.<br>5. Buat file konfigurasi GRUB<br>6. Verifikasi bahwa versi GRUB Anda mendukung tipe sistem file yang mendasari nya dan tingkatkan GRUB jika perlu.<br>7. Lepaskan lampiran sistem file.<br>8. Lampirkan ke instans asli.<br>9. Modifikasi atribut kernel untuk menggunakan citra GRUB yang sesuai (disk pertama atau partisi pertama pada disk pertama).<br>10. Mulai instans. |
| Didukung penyimpanan instans | Opsi 1: Modifikasi AMI dan luncurkan ulang instans:<br>1. Buat AMI baru dengan file konfigurasi GRUB di lokasi standar (/boot/grub/menu.lst).<br>2. Pilih citra GRUB yang sesuai, (drive hd0-1st atau hd00 – drive pertama, partisi pertama).<br>3. Verifikasi bahwa versi GRUB Anda mendukung tipe sistem file yang mendasari nya dan tingkatkan GRUB jika perlu.<br>4. Akhiri instans dan luncurkan instans baru menggunakan AMI yang telah Anda buat.<br><br>Opsi 2: Akhiri instans dan luncurkan instans baru, dengan menentukan kernel yang benar.<br><br>**Note**<br>Untuk memulihkan data dari instans yang sudah ada, hubungi Dukungan. |

# Memunculkan eth0 antarmuka: eth0 perangkat memiliki alamat MAC yang berbeda dari yang diharapkan, mengabaikan. (Alamat Hard-coded MAC)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
...
Bringing up loopback interface:  [  OK  ]

Bringing up interface eth0:  Device eth0 has different MAC address than expected,
 ignoring.
[FAILED]

Starting auditd: [  OK  ]
```

## Potensi penyebab

Terdapat MAC antarmuka dengan hard-coding dalam konfigurasi AMI

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Lakukan salah satu tindakan berikut:<br>• Modifikasi AMI untuk menghapus hard-coding dan meluncurkan ulang instans.<br>• Modifikasi instans untuk menghapus alamat MAC dengan hard-coding.<br><br>ATAU<br><br>Gunakan prosedur berikut:<br>1. Hentikan instans.<br>2. Lepaskan lampiran volume root.<br>3. Lampirkan volume ke instans lain dan modifikasi volume untuk menghapus alamat MAC dengan hard-coding. |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | 4. Lampirkan volume ke instans asli.<br>5. Mulai instans. |
| Didukung penyimpanan instans | Lakukan salah satu dari berikut ini:<br>• Modifikasi instans untuk menghapus alamat MAC dengan hard-coding.<br>• Akhiri instans dan luncurkan instans baru. |

## Tidak dapat memuat Kebijakan SELinux. Mesin berada dalam mode pemberlakuan. Menghentikan sekarang. (Kesalahan konfigurasi SELinux)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
audit(1313445102.626:2): enforcing=1 old_enforcing=0 auid=4294967295
Unable to load SELinux Policy. Machine is in enforcing mode. Halting now.
Kernel panic - not syncing: Attempted to kill init!
```

### Potensi penyebab

SELinux telah diaktifkan dengan kesalahan:

- Kernel yang disediakan tidak didukung oleh GRUB
- Kernel pengganti tidak ada

### Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Gunakan prosedur berikut:<br>1. Hentikan instans yang gagal.<br>2. Lepaskan lampiran volume root instans yang gagal. |

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| | 3. Lampirkan volume root ke instans Linux lain yang sedang berjalan (yang nantinya akan disebut sebagai instans pemulihan).<br>4. Hubungkan ke instans pemulihan dan pasang volume root dari instans yang gagal.<br>5. Nonaktifkan SELinux pada volume root yang dipasang. Proses ini bervariasi di seluruh distribusi Linux; untuk informasi lebih lanjut, lihat OS-specific dokumentasi Anda.<br><br>**Note**<br>Pada beberapa sistem, Anda menonaktifkan SELinux dengan mengatur `SELINUX=disabled` dalam file `/mount_point /etc/sysconfig/selinux` , dengan *`mount_point`* adalah lokasi tempat Anda memasang volume pada instans pemulihan.<br><br>6. Copot dan lepaskan lampiran volume root dari instans pemulihan dan lampirkan kembali ke instans asli.<br>7. Mulai instans. |
| Didukung penyimpanan instans | Gunakan prosedur berikut:<br><br>1. Akhiri instans dan luncurkan instans baru.<br>2. (Opsional) Carilah bantuan teknis untuk pemulihan data menggunakan Dukungan. |

# XENBUS: Waktu menghubungkan ke perangkat habis (waktu habis Xenbus)

Kondisi ini diindikasikan oleh log sistem yang mirip dengan yang ditunjukkan di bawah ini.

```
Linux version 2.6.16-xenU (builder@xenbat.amazonsa) (gcc version 4.0.1
20050727 (Red Hat 4.0.1-5)) #1 SMP Mon May 28 03:41:49 SAST 2007
...
XENBUS: Timeout connecting to devices!
...
Kernel panic - not syncing: No init found.  Try passing init= option to kernel.
```

## Potensi penyebab

- Perangkat blok tidak terhubung ke instans
- Instans ini menggunakan kernel instans lama

## Tindakan yang disarankan

| Untuk tipe instans ini | Lakukan hal berikut |
| --- | --- |
| Amazon EBS-backed | Lakukan salah satu tindakan berikut:<br>• Modifikasi AMI dan instans untuk menggunakan kernel modern dan luncurkan ulang instans.<br>• Boot ulang instans. |
| Didukung penyimpanan instans | Lakukan salah satu dari berikut ini:<br>• Akhiri instans.<br>• Modifikasi AMI untuk menggunakan kernel modern, dan luncurkan instans baru menggunakan AMI ini. |

# Memecahkan masalah booting instans Amazon EC2 Linux dari volume yang salah

Dalam beberapa situasi, volume selain volume yang dilampirkan ke `/dev/xvda` atau `/dev/sda` menjadi volume root dari instance Linux. Hal ini dapat terjadi ketika Anda telah melampirkan volume root dari instans lain, atau volume yang dibuat dari snapshot volume root, ke instans dengan volume root yang sudah ada.

Hal ini disebabkan oleh cara kerja ramdisk awal di Linux. Ramdisk awal di Linux memilih volume yang ditentukan sebagai `/` dalam `/etc/fstab`, dan pada beberapa distribusi, ini ditentukan oleh label yang terlampir ke partisi volume. Secara khusus, Anda menemukan bahwa `/etc/fstab` akan terlihat seperti berikut ini:

```
LABEL=/ / ext4 defaults,noatime 1 1
tmpfs /dev/shm tmpfs defaults 0 0
devpts /dev/pts devpts gid=5,mode=620 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
```

Jika Anda memeriksa label kedua volume tersebut, Anda akan melihat bahwa keduanya berisi label `/`:

```
[ec2-user ~]$ sudo e2label /dev/xvda1
/
[ec2-user ~]$ sudo e2label /dev/xvdf1
/
```

Dalam contoh ini, Anda bisa menjadi volume root yang `/dev/xvdf1` menjadi boot instans Anda setelah ramdisk awal berjalan, alih-alih `/dev/xvda1` volume dari mana Anda ingin mem-boot. Untuk menyelesaikan hal ini, gunakan perintah e2label yang sama untuk mengubah label volume terlampir yang tidak ingin Anda gunakan untuk boot.

Dalam beberapa kasus, penentuan UUID di `/etc/fstab` dapat mengatasi hal ini. Namun, jika kedua volume tersebut berasal dari snapshot yang sama, atau yang sekunder dibuat dari snapshot volume primer, keduanya memiliki UUID yang sama.

```
[ec2-user ~]$ sudo blkid
/dev/xvda1: LABEL="/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778 TYPE="ext4"
 PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-7084e28f7334
```

```
/dev/xvdf1: LABEL="old/" UUID=73947a77-ddbe-4dc7-bd8f-3fe0bc840778 TYPE="ext4"
 PARTLABEL="Linux" PARTUUID=d55925ee-72c8-41e7-b514-7084e28f7334
```

Untuk mengubah label volume ext4 terlampir

1. Gunakan perintah e2label untuk mengubah label volume agar menjadi sesuatu selain /.

   ```
   [ec2-user ~]$ sudo e2label /dev/xvdf1 old/
   ```

2. Verifikasi bahwa volume tersebut memiliki label baru.

   ```
   [ec2-user ~]$ sudo e2label /dev/xvdf1
   old/
   ```

Untuk mengubah label volume xfs terlampir

- Gunakan perintah xfs_admin untuk mengubah label volume agar menjadi sesuatu selain /.

  ```
  [ec2-user ~]$ sudo xfs_admin -L old/ /dev/xvdf1
  writing all SBs
  new label = "old/"
  ```

Setelah mengubah label volume sebagaimana ditunjukkan, Anda dapat melakukan boot ulang instans dan memiliki volume yang tepat yang dipilih oleh ramdisk awal saat instans melakukan boot.

> **Important**
>
> Jika Anda ingin melepaskan lampiran volume dengan label baru dan mengembalikannya ke instans lain untuk digunakan sebagai volume root, Anda harus melakukan prosedur di atas lagi dan mengubah label volume kembali ke nilai aslinya. Jika tidak, instans lain tidak bisa melakukan boot karena ramdisk tidak dapat menemukan volume dengan label /.

# Memecahkan masalah saat menghubungkan ke instans Windows Amazon EC2 Anda

Informasi berikut dan kesalahan umum dapat membantu Anda memecahkan masalah saat menghubungkan ke instans Windows Anda.

Masalah koneksi

- Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh
- Kesalahan menggunakan klien RDP macOS
- RDP menampilkan layar hitam, bukan desktop
- Tidak dapat masuk dari jarak jauh ke sebuah instans dengan pengguna yang bukan administrator
- Memecahkan masalah Remote Desktop menggunakan AWS Systems Manager
- Aktifkan Desktop Jarak Jauh pada instans EC2 dengan registri jarak jauh
- Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance Windows?

## Desktop Jarak Jauh tidak dapat terhubung ke komputer jarak jauh

Coba hal berikut untuk menyelesaikan masalah yang terkait dengan menghubungkan ke instans Anda:

- Verifikasi bahwa Anda menggunakan nama host DNS publik yang benar. (Di konsol Amazon EC2, pilih instans, dan periksa DNS Publik (IPv4) di panel detail.) Jika instans Anda ada di VPC dan Anda tidak melihat nama DNS publik, Anda harus mengaktifkan nama host DNS. Untuk informasi selengkapnya, lihat Atribut DNS untuk VPC Anda di Panduan Pengguna Amazon VPC.
- Verifikasi bahwa instans Anda memiliki alamat IPv4 publik. Jika tidak, Anda dapat mengaitkan alamat IP Elastis dengan instans Anda. Untuk informasi selengkapnya, lihat Alamat Elastic IP.
- Untuk terhubung ke instans Anda menggunakan alamat IPv6, periksa apakah komputer lokal Anda memiliki alamat IPv6 dan dikonfigurasi untuk menggunakan IPv6. Untuk informasi selengkapnya, lihat Konfigurasi IPv6 di instans Anda di Panduan Pengguna Amazon VPC.
- Verifikasi bahwa grup keamanan Anda memiliki aturan yang memungkinkan akses RDP pada port 3389.
- Jika Anda menyalin kata sandi, tetapi mendapatkan kesalahan `Your credentials did not work`, coba ketik secara manual saat diminta. Ada kemungkinan Anda melewatkan satu karakter atau mengetik karakter spasi tambahan saat Anda menyalin kata sandi.

- Verifikasi bahwa instans telah lulus pemeriksaan status. Untuk informasi selengkapnya, lihat Pemeriksaan status untuk instans Amazon EC2 dan the section called “Pemeriksaan status instans Linux gagal”.
- Verifikasi bahwa tabel rute untuk subnet memiliki rute yang mengirimkan semua lalu lintas yang ditujukan di luar VPC ke gateway internet untuk VPC. Untuk informasi selengkapnya, lihat Membuat tabel rute kustom (Gateway Internet) di Panduan Pengguna Amazon VPC.
- Verifikasi bahwa Windows Firewall, atau perangkat lunak firewall lainnya tidak memblokir lalu lintas RDP ke instans. Sebaiknya Anda menonaktifkan Windows Firewall dan mengendalikan akses ke instans Anda menggunakan aturan grup keamanan. Coba salah satu opsi berikut:
  - Gunakan AWSSupport-TroubleshootRDP untukdisable the Windows Firewall profiles using SSM Agent. Opsi ini mengharuskan instance dikonfigurasi untuk AWS Systems Manager.
  - Gunakan AWSSupport-ExecuteEC2Rescue.
  - Hentikan instance, lepaskan volume root, dan lampirkan volume ke instance sementara sebagai volume data. Hubungkan ke instance sementara dan bawa volume online. Muat sarang registri dari volume data. Di bawah`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\SharedAccess\Parameters\FirewallPolicyStandardProfile`, atur `EnableFirewall` ke 0. Bongkar sarang registri dan kemudian offline volume. Lepaskan volume dari instance sementara dan lampirkan ke instance asli sebagai volume root. Mulai contoh asli.
- Verifikasi bahwa Autentikasi Tingkat Jaringan dinonaktifkan pada instans yang bukan bagian dari domain Direktori Aktif (gunakan AWSSupport-TroubleshootRDP untuk disable NLA).
- Verifikasi bahwa Jenis Startup Layanan Desktop Jarak Jauh (TermService) adalah Otomatis dan layanan dimulai (gunakan AWSSupport-TroubleshootRDP untukenable and start the RDP service).
- Verifikasi bahwa Anda tersambung ke port Remote Desktop Protocol yang benar, yang secara default adalah 3389 (gunakan AWSSupport-TroubleshootRDP untuk read the current RDP port dan change it back to 3389).
- Verifikasi bahwa koneksi Desktop Jarak Jauh diizinkan pada instans Anda (gunakan AWSSupport-TroubleshootRDP ke enable Remote Desktop connections).
- Verifikasi bahwa kata sandi belum kedaluwarsa. Jika kata sandi telah kedaluwarsa, Anda dapat mengatur ulang kata sandi. Untuk informasi selengkapnya, lihat Setel ulang kata sandi administrator Windows untuk instans Windows Amazon EC2.
- Jika Anda mencoba untuk terhubung menggunakan pengguna yang dibuat pada instans dan menerima kesalahan `The user cannot connect to the server due to insufficient access privileges`, verifikasi bahwa Anda memberi pengguna hak untuk masuk secara lokal. Untuk informasi selengkapnya, lihat Memberi Anggota Hak untuk Masuk secara Lokal.

- Jika Anda mencoba lebih dari sesi RDP bersamaan maksimum yang diperbolehkan, sesi Anda akan diakhiri dengan pesan `Your Remote Desktop Services session has ended. Another user connected to the remote computer, so your connection was lost.` Secara default, Anda mengizinkan dua sesi RDP bersamaan untuk instans Anda.

## Kesalahan menggunakan klien RDP macOS

Jika Anda menghubungkan ke instance Windows Server menggunakan klien Remote Desktop Connection dari situs web Microsoft, Anda mungkin mendapatkan kesalahan berikut:

```
Remote Desktop Connection cannot verify the identity of the computer that you want to
 connect to.
```

Unduh aplikasi Microsoft Remote Desktop dari Mac App Store dan gunakan aplikasi untuk terhubung ke instans Anda.

## RDP menampilkan layar hitam, bukan desktop

Untuk mengatasi masalah ini, coba hal berikut:

- Periksa output konsol untuk informasi tambahan. Untuk mendapatkan output konsol pada instans Anda menggunakan konsol Amazon EC2, pilih instans, lalu pilih Tindakan, Pantau dan atasi masalah, Dapatkan log sistem.
- Verifikasi bahwa Anda menjalankan versi terbaru klien RDP Anda.
- Coba pengaturan default untuk klien RDP.
- Jika Anda menggunakan Koneksi Desktop Jarak Jauh, coba mulai dengan opsi `/admin` sebagai berikut.

  ```
  mstsc /v:instance /admin
  ```

- Jika server menjalankan aplikasi layar penuh, server mungkin berhenti merespons. Gunakan Ctrl +Shift+ Esc untuk memulai Windows Task Manager, lalu tutup aplikasi.
- Jika server terlalu banyak digunakan, mungkin sudah berhenti merespons. Untuk memantau instans menggunakan konsol Amazon EC2, pilih instans, lalu pilih tab Pemantauan. Jika Anda perlu mengubah tipe instans ke ukuran yang lebih besar, lihat Perubahan jenis instans Amazon EC2.

# Tidak dapat masuk dari jarak jauh ke sebuah instans dengan pengguna yang bukan administrator

Jika Anda tidak dapat masuk dari jarak jauh ke instans Windows dengan pengguna yang bukan akun administrator, pastikan bahwa Anda telah memberi pengguna hak untuk masuk secara lokal. Lihat Memberi pengguna atau grup hak untuk masuk secara lokal ke pengendali domain di domain.

# Memecahkan masalah Remote Desktop menggunakan AWS Systems Manager

Anda dapat menggunakan AWS Systems Manager untuk memecahkan masalah yang menghubungkan ke instance Windows Anda menggunakan RDP.

## AWSSupport-TroubleshootRDP

Dok AWSSupport-TroubleshootRDP umen otomatisasi memungkinkan pengguna untuk memeriksa atau memodifikasi pengaturan umum pada instance target yang dapat memengaruhi koneksi Remote Desktop Protocol (RDP), seperti Port  RDP,  Network Layer Authentication (NLA), dan profil Windows Firewall.  Secara default, dokumen membaca dan mengeluarkan nilai pengaturan ini.

Dok AWSSupport-TroubleshootRDP umen otomatisasi dapat digunakan dengan instans EC2, instans lokal, dan mesin virtual (VM) yang diaktifkan untuk digunakan dengan AWS Systems Manager (instans terkelola). Selain itu, dokumen otomatisasi juga dapat digunakan dengan instans EC2 pada Windows Server yang tidak diaktifkan untuk digunakan dengan Systems Manager. Untuk informasi tentang mengaktifkan instance untuk digunakan dengan AWS Systems Manager, lihat N ode ter kelola di Panduan AWS Systems Manager Pengguna.

Untuk memecahkan masalah menggunakan AWSSupport-TroubleshootRDP dokumen

1. Masuk ke Konsol Systems Manager.
2. Verifikasi bahwa Anda berada di Wilayah yang sama dengan instans yang mengalami gangguan.
3. Pilih Dokumen dari panel navigasi kiri.
4. Pada tab Dimiliki oleh Amazon, masukkan `AWSSupport-TroubleshootRDP` di bidang pencarian. Saat dokumen `AWSSupport-TroubleshootRDP` muncul, pilihlah.
5. Pilih Eksekusi otomatisasi.
6. Untuk Mode Eksekusi, pilih Eksekusi sederhana.

7. Untuk parameter Input,InstanceId, aktifkan Tampilkan alat pilih instance interaktif.
8. Pilih instans Amazon EC2 Anda.
9. Tinjau contoh, lalu pilih Eksekusi.
10. Untuk memantau kemajuan eksekusi, pada Status eksekusi, tunggu status berubah dari Tertunda menjadi Berhasil. Perluas Output untuk melihat hasilnya. Untuk melihat output setiap langkah, di Langkah-langkah yang Dieksekusi, pilih item dari ID Langkah.

### AWSSupport-TroubleshootRDP contoh

Contoh berikut menunjukkan cara menyelesaikan tugas pemecahan masalah umum menggunakanAWSSupport-TroubleshootRDP. Anda dapat menggunakan AWS CLI start-automation-execution perintah contoh atau tautan yang disediakan ke Konsol Manajemen AWS.

#### Example Contoh: Periksa status RDP saat ini

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --
parameters "InstanceId=i-1234567890abcdef0, Action=Custom" --region region_code
```

AWS Systems Manager konsol:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region#documentVersion=$LATEST
```

#### Example Contoh: Nonaktifkan Windows Firewall

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP"
 --parameters "InstanceId=i-1234567890abcdef0, Action=Custom, Firewall=Disable" --
region region_code
```

AWS Systems Manager konsol:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion=$LATEST&Firewall=Disable
```

Example Contoh: Nonaktifkan Autentikasi Tingkat Jaringan

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --
parameters "InstanceId=i-1234567890abcdef0, Action=Custom, NLASettingAction=Disable" --
region region_code
```

AWS Systems Manager konsol:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion
```

Example Contoh: Atur Tipe Startup Layanan RDP ke Otomatis dan mulai layanan RDP

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --
parameters "InstanceId=i-1234567890abcdef0, Action=Custom, RDPServiceStartupType=Auto,
 RDPServiceAction=Start" --region region_code
```

AWS Systems Manager konsol:

```
https://console.aws.amazon.com/systems-manager/automation/execute/
AWSSupport-TroubleshootRDP?region=region_code#documentVersion=
$LATEST&RDPServiceStartupType=Auto&RDPServiceAction=Start
```

Example Contoh: Pulihkan Port RDP default (3389)

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --
parameters "InstanceId=i-1234567890abcdef0, Action=Custom, RDPPortAction=Modify" --
region region_code
```

AWS Systems Manager konsol:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion=$LATEST&RDPPortAction=Modify
```

Example Contoh: Izinkan koneksi jarak jauh

AWS CLI:

```
aws ssm start-automation-execution --document-name "AWSSupport-TroubleshootRDP" --
parameters "InstanceId=i-1234567890abcdef0, Action=Custom, RemoteConnections=Enable" --
region region_code
```

AWS Systems Manager konsol:

```
https://console.aws.amazon.com/systems-manager/automation/execute/AWSSupport-
TroubleshootRDP?region=region_code#documentVersion=$LATEST&RemoteConnections=Enable
```

## AWSSupport-ExecuteEC2Rescue

Dokumen AWSSupport-ExecuteEC2Rescue otomatisasi menggunakan EC2Rescue untuk Windows Server untuk secara otomatis memecahkan masalah dan memulihkan konektivitas instance EC2 dan masalah RDP. Untuk informasi selengkapnya, lihat Jalankan alat EC2Rescue di instans yang tak terjangkau.

Dok AWSSupport-ExecuteEC2Rescue umen otomatisasi memerlukan penghentian dan restart instance. Otomatisasi System Manager menghentikan instans dan membuat Amazon Machine Image (AMI). Data yang disimpan dalam volume penyimpanan instans hilang. Alamat IP publik berubah jika Anda tidak menggunakan alamat IP Elastis. Untuk informasi selengkapnya, lihat Jalankan alat EC2Rescue di instans yang tak terjangkau di Panduan Pengguna AWS Systems Manager .

Untuk memecahkan masalah menggunakan AWSSupport-ExecuteEC2Rescue dokumen

1. Buka konsol System Manager.
2. Verifikasi bahwa Anda berada di Wilayah yang sama dengan instans Amazon EC2 yang mengalami gangguan.
3. Di panel navigasi, pilih Dokumen.
4. Cari dan pilih dokumen `AWSSupport-ExecuteEC2Rescue`, lalu pilih Eksekusi otomatisasi.
5. Untuk Mode Eksekusi, pilih Eksekusi sederhana.
6. Di bagian Parameter masukan, untukUnreachableInstanceId, masukkan ID instans Amazon EC2 dari instans yang tidak dapat dijangkau.

7. (Opsional) UntukLogDestination, masukkan nama bucket Amazon Simple Storage Service (Amazon S3) jika Anda ingin mengumpulkan log sistem operasi untuk memecahkan masalah instans Amazon EC2 Anda. Log secara otomatis diunggah ke bucket yang ditentukan.
8. Pilih Eksekusi.
9. Untuk memantau kemajuan eksekusi, pada Status eksekusi, tunggu status berubah dari Tertunda menjadi Berhasil. Perluas Output untuk melihat hasilnya. Untuk melihat output setiap langkah, di Langkah-langkah yang Dieksekusi, pilih ID Langkah.

## Aktifkan Desktop Jarak Jauh pada instans EC2 dengan registri jarak jauh

Jika instans yang tidak dapat dijangkau tidak dikelola oleh AWS Manajer Sesi Sistem, maka Anda dapat menggunakan registri jarak jauh untuk mengaktifkan Remote Desktop.

1. Dari konsol EC2, hentikan instans yang tak terjangkau.
2. Lepaskan volume root dari instans tak terjangkau dan lampirkan ke instans terjangkau di Zona Ketersediaan yang sama dengan volume penyimpanan. Jika Anda tidak memiliki instans terjangkau di Zona Ketersediaan yang sama, luncurkan satu instans. Perhatikan nama perangkat volume root pada instans yang tak terjangkau.
3. Pada instans terjangkau, buka Manajemen Disk. Anda dapat melakukannya dengan menjalankan perintah berikut di jendela Prompt Perintah.

   ```
   diskmgmt.msc
   ```

4. Buka menu konteks (klik kanan) untuk volume yang baru dilampirkan yang berasal dari instance yang tidak dapat dijangkau, lalu pilih  Online.
5. Buka Windows Registry Editor. Anda dapat melakukannya dengan menjalankan perintah berikut di jendela Prompt Perintah.

   ```
   regedit
   ```

6. Di Registry Editor, pilih HKEY_LOCAL_MACHINE, lalu pilih File, Muat Hive.
7. Pilih drive dari volume yang terlampir, navigasikan ke `\Windows\System32\config\`, pilih `SYSTEM`, lalu pilih Buka.
8. Untuk Nama Kunci, masukkan nama unik untuk hive dan pilih OKE.
9. Cadangkan hive registri sebelum membuat perubahan pada registri tersebut.

a. Di pohon konsol Editor Registri, pilih sarang yang Anda muat: HKEY_LOCAL_ MACHINE\. *your-key-name*

b. Pilih File, Ekspor.

c. Di kotak dialog Ekspor File Registri, pilih lokasi tempat Anda ingin menyimpan salinan cadangan, lalu ketik nama untuk file cadangan di bidang Nama file.

d. Pilih Simpan.

10. Di Registry Editor, arahkan keHKEY_LOCAL_MACHINE*your key name*\ControlSet001\Control\Terminal Server, dan kemudian, di panel rincian, buka FDen yTSConnections.
11. Di kotak nilai Edit DWORD, masukkan `0` di bidang Data nilai.
12. Pilih OKE.

> **Note**
>
> Jika nilai di bidang Data nilai adalah `1`, instans akan menolak koneksi desktop jarak jauh. Nilai `0` memungkinkan koneksi desktop jarak jauh.

13. Di Editor Registri, pilih HKEY_LOCAL_MACHINE*your-key-name*, lalu pilih File, Unload Hive.
14. Tutup Registry Editor dan Manajemen Disk.
15. Dari konsol EC2, lepaskan volume dari instans terjangkau, dan lampirkan kembali ke instans yang tak terjangkau. Saat melampirkan volume ke instans tak terjangkau, masukkan nama perangkat yang Anda simpan sebelumnya di bidang perangkat.
16. Mulai ulang instans tak terjangkau.

## Saya kehilangan kunci privat. Bagaimana saya bisa terhubung ke instance Windows?

Saat Anda terhubung ke instans Windows yang baru diluncurkan, Anda mendekripsi kata sandi untuk akun Administrator menggunakan kunci privat untuk pasangan kunci yang Anda tentukan saat meluncurkan instans.

Jika Anda menghilangkan kata sandi Administrator dan tidak lagi memiliki kunci privat, Anda harus mengatur ulang kata sandi atau membuat sebuah instans baru. Untuk informasi selengkapnya, lihat Setel ulang kata sandi administrator Windows untuk instans Windows Amazon EC2. Untuk langkah-

langkah mengatur ulang kata sandi menggunakan dokumen Systems Manager, lihat Atur ulang kata sandi dan kunci SSH pada instans EC2 di Panduan Pengguna AWS Systems Manager .

# Memecahkan masalah mulai instans Amazon EC2 Windows

Berikut ini adalah kiat pemecahan masalah untuk membantu Anda memecahkan masalah kata sandi dan aktivasi dengan instans Amazon EC2 Windows.

Masalah

- "Kata sandi tidak tersedia"
- "Kata sandi belum tersedia"
- "Tidak dapat mengambil kata sandi Windows"
- "Menunggu layanan metadata"
- "Tidak dapat mengaktifkan Windows"
- "Windows tidak asli (0x80070005)"
- "Tidak ada Server Lisensi Server Terminal yang tersedia untuk memberikan lisensi"
- "Beberapa pengaturan dikelola oleh organisasi Anda"

## "Kata sandi tidak tersedia"

Untuk menghubungkan ke instans Windows menggunakan Desktop Jarak Jauh, Anda harus menentukan akun dan kata sandi. Akun dan kata sandi yang diberikan didasarkan pada AMI yang Anda gunakan untuk meluncurkan instans. Anda dapat mengambil kata sandi yang dibuat secara otomatis untuk akun Administrator, atau menggunakan akun dan kata sandi yang digunakan dalam instans asli tempat AMI dibuat.

Anda dapat membuat kata sandi untuk akun Administrator pada instans yang diluncurkan menggunakan AMI Windows kustom. Untuk menghasilkan kata sandi, Anda perlu mengonfigurasi beberapa pengaturan di sistem operasi sebelum AMI dibuat. Untuk informasi selengkapnya, lihat Buat EBS-backed AMI Amazon.

Jika instans Windows Anda tidak dikonfigurasi untuk menghasilkan kata sandi acak, Anda akan menerima pesan berikut saat Anda mengambil kata sandi yang dibuat secara otomatis menggunakan konsol:

```
Password is not available.
```

```
The instance was launched from a custom AMI, or the default password has changed. A
password cannot be retrieved for this instance. If you have forgotten your password,
 you can
reset it using the Amazon EC2 configuration service. For more information, see
 Passwords for a
Windows Server instance.
```

Periksa output konsol pada instans tersebut untuk melihat apakah AMI yang Anda gunakan untuk meluncurkannya dibuat dengan pembuatan kata sandi yang nonaktif. Jika pembuatan kata sandi dinonaktifkan, output konsol berisi hal-hal berikut ini:

```
Ec2SetPassword: Disabled
```

Jika pembuatan kata sandi dinonaktifkan dan Anda tidak ingat kata sandi untuk instans aslinya, Anda dapat menyetel ulang kata sandi untuk instans ini. Untuk informasi selengkapnya, lihat Setel ulang kata sandi administrator Windows untuk instans Windows Amazon EC2.

## “Kata sandi belum tersedia”

Untuk menghubungkan ke instans Windows menggunakan Desktop Jarak Jauh, Anda harus menentukan akun dan kata sandi. Akun dan kata sandi yang diberikan didasarkan pada AMI yang Anda gunakan untuk meluncurkan instans. Anda dapat mengambil kata sandi yang dibuat secara otomatis untuk akun Administrator, atau menggunakan akun dan kata sandi yang digunakan dalam instans asli tempat AMI dibuat.

Kata sandi Anda akan tersedia dalam beberapa menit. Jika kata sandi tidak tersedia, Anda akan menerima pesan berikut ketika Anda mengambil kata sandi yang dibuat secara otomatis menggunakan konsol:

```
Password not available yet.
Please wait at least 4 minutes after launching an instance before trying to retrieve
 the
auto-generated password.
```

Jika sudah lebih dari empat menit dan Anda masih tidak bisa mendapatkan kata sandinya, mungkin agen peluncuran untuk instans Anda tidak dikonfigurasi untuk membuat kata sandinya. Verifikasi dengan memeriksa apakah output konsol kosong atau tidak. Untuk informasi selengkapnya, lihat Tidak bisa mendapatkan output konsol.

Juga verifikasi bahwa akun AWS Identity and Access Management (IAM) yang digunakan untuk mengakses Portal Manajemen memiliki `ec2:GetPasswordData` tindakan yang diizinkan. Untuk informasi tentang izin IAM selengkapnya, lihat Apa itu IAM?.

## “Tidak dapat mengambil kata sandi Windows”

Untuk mendapatkan kembali kata sandi yang dibuat secara otomatis pada akun Administrator, Anda harus menggunakan kunci privat untuk pasangan kunci yang Anda tentukan saat meluncurkan instans. Jika Anda tidak menentukan pasangan kunci saat meluncurkan instans, Anda akan menerima pesan berikut.

```
Cannot retrieve Windows password
```

Anda dapat menghentikan instans ini dan meluncurkan instans baru menggunakan AMI yang sama, pastikan untuk menentukan pasangan kunci.

## “Menunggu layanan metadata”

Sebelum diaktifkan, instans Windows harus memperoleh informasi dari metadata instans miliknya. Secara default, pengaturan `WaitForMetaDataAvailable` memastikan bahwa layanan EC2Config menunggu metadata instans dapat diakses sebelum melanjutkan proses boot. Untuk informasi selengkapnya, lihat Menggunakan metadata instans untuk mengelola instans EC2 Anda.

Jika instans gagal dalam uji jangkauan instans, coba langkah berikut untuk menyelesaikan masalah ini.

- Periksa blok CIDR untuk VPC Anda. Instans Windows tidak dapat melakukan boot dengan benar jika diluncurkan ke VPC yang memiliki rentang alamat IP dari `224.0.0.0` hingga `255.255.255.255` (Rentang Alamat IP Kelas D dan Kelas E). Rentang alamat IP ini disimpan, dan tidak boleh ditetapkan ke perangkat host. Sebaiknya Anda membuat VPC dengan blok CIDR dari rentang alamat IP privat (tidak dapat dirutekan secara publik) seperti yang ditentukan dalam RFC 1918.
- Mungkin saja sistem telah dikonfigurasi dengan alamat IP statis. Coba buat antarmuka jaringan dan lampirkan ke instans.
- Untuk mengaktifkan DHCP pada instans Windows yang tidak dapat Anda sambungkan

  1. Hentikan instans yang terpengaruh dan lepaskan volume root-nya.

2. Luncurkan instans sementara di Zona Ketersediaan yang sama dengan instans yang terpengaruh.

> **Warning**
>
> Jika instans sementara Anda didasarkan pada AMI yang sama dengan instans asli, Anda harus menyelesaikan langkah-langkah tambahan atau Anda tidak akan dapat melakukan boot instans asli setelah memulihkan volume root-nya karena tabrakan tanda tangan disk. Atau, pilih AMI yang berbeda untuk instans sementara. Misalnya, jika instance asli menggunakan AWS Windows AMI untuk Windows Server 2016, luncurkan instance sementara menggunakan AWS Windows AMI untuk Windows Server 2019.

3. Lampirkan volume root dari instans yang terpengaruh ke instans sementara ini. Hubungkan ke instans sementara, buka utilitas Manajemen Disk, dan buat drive menjadi online.
4. Dari instans sementara, buka Regedit dan pilih HKEY_LOCAL_MACHINE. Dari menu File, pilih Muat Hive. Pilih drive, buka file `Windows\System32\config\SYSTEM`, dan tentukan nama kunci saat diminta (Anda dapat menggunakan nama apa pun).
5. Pilih kunci yang baru saja Anda muat dan navigasikan ke `ControlSet001\Services\Tcpip\Parameters\Interfaces`. Setiap antarmuka jaringan dicantumkan oleh GUID. Pilih antarmuka jaringan yang benar. Jika DHCP dinonaktifkan dan alamat IP statis ditetapkan, `EnableDHCP` diatur ke 0. Untuk mengaktifkan DHCP, atur `EnableDHCP` ke 1, dan hapus kunci berikut jika ada: `NameServer`, `SubnetMask`, `IPAddress`, dan `DefaultGateway`. Pilih lagi kunci tersebut, dan dari menu File, pilih Bongkar Hive.

> **Note**
>
> Jika Anda memiliki banyak antarmuka jaringan, Anda harus mengidentifikasi antarmuka yang benar untuk mengaktifkan DHCP. Untuk mengidentifikasi antarmuka jaringan yang benar, tinjau nilai kunci berikut `NameServer`, `SubnetMask`, `IPAddress`, dan `DefaultGateway`. Nilai-nilai ini menampilkan konfigurasi statis instans sebelumnya.

6. (Opsional) Jika DHCP sudah diaktifkan, Anda mungkin tidak memiliki rute ke layanan metadata. Memperbarui EC2Config dapat mengatasi masalah ini.

a. Unduh dan instal versi terbaru layanan EC2Config. Untuk informasi selengkapnya tentang menginstal layanan ini, lihat the section called “Instal EC2Config”.

b. Ekstrak file dari file `.zip` ke direktori `Temp` pada drive yang Anda lampirkan.

c. Buka Regedit dan pilih HKEY_LOCAL_MACHINE. Dari menu File, pilih Muat Hive. Pilih drive, buka file `Windows\System32\config\SOFTWARE`, dan tentukan nama kunci saat diminta (Anda dapat menggunakan nama apa pun).

d. Pilih kunci yang baru saja Anda muat dan navigasikan ke `Microsoft\Windows \CurrentVersion`. Pilih kunci `RunOnce`. (Jika kunci ini tidak ada, klik kanan `CurrentVersion`, arahkan ke Baru, pilih Kunci, dan beri nama kunci `RunOnce`.) Buka menu konteks (klik kanan), arahkan ke Baru, dan pilih Nilai String. Masukkan`Ec2Install` sebagai nama dan `C:\Temp\Ec2Install.exe -q` sebagai data.

e. Pilih lagi kunci tersebut, dan dari menu File, pilih Bongkar Hive.

7. (Opsional) Jika instans sementara Anda didasarkan pada AMI yang sama dengan instans asli, Anda harus menyelesaikan langkah-langkah berikut atau Anda tidak akan dapat melakukan boot instans asli setelah memulihkan volume root-nya karena tabrakan tanda tangan disk.

> **Warning**
>
> Prosedur berikut menjelaskan cara mengedit Windows Registry menggunakan Registry Editor. Jika Anda tidak terbiasa dengan Windows Registry atau cara membuat perubahan dengan aman menggunakan Registry Editor, lihat Konfigurasi Registry.

a. Buka prompt perintah, ketik regedit.exe, dan tekan Enter.

b. Di Registry Editor, pilih HKEY_LOCAL_MACHINE dari menu konteks (klik kanan), lalu pilih Temukan.

c. Ketik Windows Boot Manager, lalu pilih Temukan Berikutnya.

d. Pilih kunci bernama `11000001`. Kunci ini adalah kelompok dari kunci yang Anda temukan di langkah sebelumnya.

e. Pada panel kanan, pilih `Element`, lalu pilih Ubah dari menu konteks (klik kanan).

f. Temukan tanda tangan disk empat bita pada offset 0x38 dalam data tersebut. Balikkan bita tersebut untuk membuat tanda tangan disk, dan tuliskan. Misalnya, tanda tangan disk yang diwakili oleh data berikut ini adalah `E9EB3AA5`:

```
...
0030  00 00 00 00 01 00 00 00
0038  A5 3A EB E9 00 00 00 00
0040  00 00 00 00 00 00 00 00
...
```

g. Di jendela Command Prompt, jalankan perintah berikut untuk memulai Microsoft DiskPart.

```
diskpart
```

h. Jalankan DiskPart perintah berikut untuk memilih volume. (Anda dapat memverifikasi bahwa nomor disk adalah 1 menggunakan utilitas Manajemen Disk.)

```
DISKPART> select disk 1

Disk 1 is now the selected disk.
```

i. Jalankan DiskPart perintah berikut untuk mendapatkan tanda tangan disk.

```
DISKPART>  uniqueid disk

Disk ID: 0C764FA8
```

j. Jika tanda tangan disk yang ditampilkan pada langkah sebelumnya tidak cocok dengan tanda tangan disk dari BCD yang Anda tulis sebelumnya, gunakan DiskPart perintah berikut untuk mengubah tanda tangan disk sehingga cocok:

```
DISKPART> uniqueid disk id=E9EB3AA5
```

8. Menggunakan utilitas Manajemen Disk, buat drive menjadi offline.

> **Note**
>
> Drive secara otomatis offline jika instans sementara menjalankan sistem operasi yang sama dengan instans yang terpengaruh, jadi Anda tidak perlu membuatnya offline secara manual.

9. Lepaskan volume dari instans sementara. Anda dapat mengakhiri instans sementara jika tidak menggunakannya lagi.

10. Pulihkan volume root dari instans yang terpengaruh dengan melampirkan volume sebagai `/dev/sda1`.
11. Mulai instans yang terpengaruh.

Jika Anda terhubung ke instans, buka peramban Internet dari instans dan masukkan URL berikut untuk server metadata:

```
http://169.254.169.254/latest/meta-data/
```

Jika Anda tidak dapat menghubungi server metadata, coba langkah berikut ini untuk menyelesaikan masalah ini:

- Unduh dan instal versi terbaru layanan EC2Config. Untuk informasi selengkapnya tentang menginstal layanan ini, lihat the section called “Instal EC2Config”.
- Periksa apakah instance Windows menjalankan driver Red Hat PV. Jika iya, perbarui ke driver Citrix PV. Untuk informasi selengkapnya, lihat the section called “Mutakhirkan driver PV”.
- Verifikasi bahwa firewall, IPSec, dan pengaturan proksi tidak memblokir lalu lintas keluar ke layanan metadata (`169.254.169.254`) atau server AWS KMS (alamat ditentukan di elemen `TargetKMSServer` di `C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml`).
- Pastikan Anda memiliki rute ke layanan metadata (`169.254.169.254`) menggunakan perintah berikut.

  ```
  route print
  ```

- Periksa masalah jaringan yang mungkin memengaruhi Zona Ketersediaan untuk instans Anda. Kunjungi http://status.aws.amazon.com/.

## “Tidak dapat mengaktifkan Windows”

Instans Windows menggunakan AWS KMS aktivasi Windows. Anda dapat menerima pesan ini:`A problem occurred when Windows tried to activate. Error Code 0xC004F074`, jika instans Anda tidak dapat mencapai AWS KMS server. Windows harus diaktifkan setiap 180 hari. EC2config mencoba menghubungi AWS KMS server sebelum periode aktivasi berakhir untuk memastikan bahwa Windows tetap diaktifkan.

Jika Anda mengalami masalah aktivasi Windows, gunakan prosedur berikut ini untuk menyelesaikan masalah tersebut.

Untuk EC2Config (AMI Windows Server 2012 R2 dan versi sebelumnya)

1. Unduh dan instal versi terbaru layanan EC2Config. Untuk informasi selengkapnya tentang menginstal layanan ini, lihat the section called “Instal EC2Config”.
2. Masuk ke instans dan buka file berikut: `C:\Program Files\Amazon\Ec2ConfigService\Settings\config.xml`.
3. TemukanEc2WindowsActivate plugin di`config.xml` mengajukan. Ubah statusnya ke Diaktifkan dan simpan perubahan Anda.
4. Di snap-in Windows Services, mulai ulang layanan EC2Config atau boot ulang instans.

Jika langkah ini tidak menyelesaikan masalah aktivasi, ikuti langkah-langkah tambahan berikut.

1. Tetapkan AWS KMS target: C:\> slmgr.vbs /skms 169.254.169.250:1688
2. Aktifkan Windows: C:\> slmgr.vbs /ato

Untuk EC2Launch (AMI Windows Server 2016 dan yang lebih baru)

1. Dari PowerShell prompt dengan hak administratif, impor modul EC2launch:

```
PS C:\> Import-Module "C:\ProgramData\Amazon\EC2-Windows\Launch\Module
\Ec2Launch.psd1"
```

2. Panggil Add-Routes fungsi untuk melihat daftar rute baru:

```
PS C:\> Add-Routes
```

3. Panggil fung Set-ActivationSettings sinya:

```
PS C:\> Set-Activationsettings
```

4. Kemudian, jalankan script berikut untuk mengaktifkan Windows:

```
PS C:\> cscript "${env:SYSTEMROOT}\system32\slmgr.vbs" /ato
```

Baik untuk EC2Config maupun EC2Launch, jika Anda masih menerima kesalahan aktivasi, verifikasi informasi berikut.

- Pastikan Anda memiliki rute ke AWS KMS server. Buka `C:\Program Files\Amazon\Ec2ConfigService\Settings\ActivationSettings.xml` dan temukan elemen `TargetKMSServer`. Jalankan perintah berikut dan periksa apakah alamat untuk AWS KMS server ini terdaftar.

  ```
  route print
  ```

- Pastikan kunci AWS KMS klien diatur. Jalankan perintah berikut dan periksa output-nya.

  ```
  C:\Windows\System32\slmgr.vbs /dlv
  ```

  Jika output berisi Kes alahan: kunci produk tidak ditemukan, kunci AWS KMS klien tidak disetel. Jika kunci AWS KMS klien tidak disetel, cari kunci klien seperti yang dijelaskan dalam artikel Microsoft ini: aktivasi AWS KMS klien dan kunci produk, lalu jalankan perintah berikut untuk mengatur kunci AWS KMS klien.

  ```
  C:\Windows\System32\slmgr.vbs /ipk client_key
  ```

- Pastikan sistem memiliki waktu dan zona waktu yang benar. Jika Anda menggunakan zona waktu selain UTC, tambahkan kunci registri berikut dan atur 1 untuk memastikan bahwa waktunya benar:`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\TimeZoneInformation\RealTimeIsUniversal`.
- Jika Windows Firewall diaktifkan, nonaktifkan untuk sementara menggunakan perintah berikut.

  ```
  netsh advfirewall set allprofiles state off
  ```

## “Windows tidak asli (0x80070005)”

Instans Windows menggunakan AWS KMS aktivasi Windows. Jika sebuah instans tidak dapat menyelesaikan proses aktivasi, instans akan melaporkan bahwa salinan Windows tidak asli.

Coba saran untuk “Tidak dapat mengaktifkan Windows”.

## “Tidak ada Server Lisensi Server Terminal yang tersedia untuk memberikan lisensi”

Secara default, Windows Server dilisensikan untuk dua pengguna simultan melalui Desktop Jarak Jauh. Jika Anda perlu memberi lebih dari dua pengguna akses simultan ke instans Windows melalui Desktop Jarak Jauh, Anda dapat membeli lisensi akses klien (CAL) Remote Desktop Services dan menginstal peran Host Sesi Desktop Jarak Jauh dan Server Lisensi Desktop Jarak Jauh.

Periksa masalah berikut:

- Anda telah melebihi jumlah maksimum sesi RDP bersamaan.
- Anda telah menginstal peran Layanan Windows Remote Desktop Services.
- Lisensi telah kedaluwarsa. Jika lisensi telah kedaluwarsa, Anda tidak dapat terhubung ke instans Windows Anda sebagai pengguna. Anda dapat mencoba hal-hal berikut ini:
  - Hubungkan ke instans dari baris perintah menggunakan parameter `/admin`, misalnya:

    ```
    mstsc /v:instance /admin
    ```

    Untuk informasi selengkapnya, lihat artikel Microsoft berikut: Akses Desktop Jarak Jauh Melalui Baris Perintah.
  - Hentikan instans, lepaskan volume Amazon EBS-nya, dan lampirkan ke instans lain di Zona Ketersediaan yang sama untuk memulihkan data Anda.

## “Beberapa pengaturan dikelola oleh organisasi Anda”

Instans yang diluncurkan dari AMI Windows Server terbaru mungkin menampilkan pesan dialog Pembaruan Windows yang menyatakan “Beberapa pengaturan dikelola oleh organisasi Anda”. Pesan ini muncul sebagai akibat dari perubahan di Windows Server dan tidak memengaruhi perilaku Pembaruan Windows atau kemampuan Anda untuk mengelola pengaturan pembaruan.

Untuk menghapus peringatan

1. Buka `gpedit.msc` dan arahkan ke Konfigurasi Komputer, Templat Administratif, Komponen Windows, Pembaruan Windows. Edit Konfigurasi Pembaruan Otomatis, dan atur ke aktif.
2. Di perintah prompt, perbarui kebijakan grup menggunakan gpupdate /force.

3. Tutup dan buka kembali Pengaturan Pembaruan Windows. Anda akan melihat pesan di atas tentang pengaturan yang dikelola oleh organisasi Anda, diikuti dengan “Kami akan mengunduh pembaruan secara otomatis, kecuali pada koneksi terukur (di mana biaya dapat berlaku). Dalam hal ini, kami akan mengunduh pembaruan yang diperlukan secara otomatis agar Windows tetap berjalan dengan lancar”.
4. Kembali ke `gpedit.msc` dan atur kebijakan grup kembali ke tidak dikonfigurasi. Jalankan lagi gpupdate /force.
5. Tutup perintah prompt dan tunggu beberapa menit.
6. Buka kembali Pengaturan Pembaruan Windows. Anda tidak akan melihat pesan “Beberapa pengaturan dikelola oleh organisasi Anda”.

# Memecahkan masalah dengan instans Amazon EC2 Windows

Berikut ini adalah kiat pemecahan masalah untuk membantu Anda memecahkan masalah dengan instans Windows Amazon EC2.

Masalah

- Tidak dapat terhubung AWS Systems Manager Manajer Sesi ke instans Windows Server 2025
- Volume EBS tidak diinisialisasi di Windows Server 2016 dan 2019
- Lakukan boot instans Windows EC2 ke Directory Service Restore Mode (DSRM)
- Instans kehilangan konektivitas jaringan atau tugas terjadwal tidak berjalan saat diharapkan
- Tidak bisa mendapatkan output konsol
- Windows Server 2012 R2 tidak tersedia di jaringan
- Tabrakan tanda tangan disk

## Tidak dapat terhubung AWS Systems Manager Manajer Sesi ke instans Windows Server 2025

Anda mungkin mengalami masalah menghubungkan AWS Systems Manager Sessions Manager ke instance Windows Server 2025. Untuk mengatasi masalah ini, masuk ke instance, lalu arahkan ke`Settings > Apps > Optional Features`, dan tambahkan`WMIC`. Mulai ulang layanan Agen SSM atau reboot instance, dan Sessions Manager harus terhubung.

Anda juga dapat menggunakan PowerShell perintah berikut untuk melakukan tindakan yang sama:

```
Start-Process -FilePath "$env:SystemRoot\system32\Dism.exe" -ArgumentList @('/
Online', '/Add-Capability', '/CapabilityName:WMIC~~~~') -Wait; Restart-Service -Name
 AmazonSSMAgent
```

## Volume EBS tidak diinisialisasi di Windows Server 2016 dan 2019

Instans yang dibuat dari Amazon Machine Images (AMI) untuk Windows Server 2016 dan 2019 menggunakan agen EC2Launch v1 pada berbagai tugas startup, termasuk menginisialisasi volume EBS. Secara default, EC2Launch v1 tidak menginisialisasi volume sekunder. Namun, Anda dapat mengonfigurasi EC2Launch v1 untuk menginisialisasi disk ini secara otomatis, sebagai berikut.

Memetakan huruf drive ke volume

1. Hubungkan ke instans untuk mengonfigurasi dan membuka file `C:\ProgramData\Amazon\EC2-Windows\Launch\Config\DriveLetterMappingConfig.json` di editor teks.
2. Tentukan pengaturan volume sebagai berikut:

   ```
   {
   "driveLetterMapping": [
    {
      "volumeName": "sample volume",
      "driveLetter": "H"
    }]
   }
   ```
3. Simpan perubahan Anda dan tutup file.
4. Buka Windows PowerShell dan gunakan perintah berikut untuk menjalankan skrip EC2launch v1 yang menginisialisasi disk:

   ```
   PS C:\>  C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1
   ```

   Untuk inisialisasi disk setiap kali melakukan boot instans, tambahkan bendera `-Schedule` sebagai berikut:

   ```
   PS C:\>  C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts\InitializeDisks.ps1 -
   Schedule
   ```

   Agen EC2Launch v1 dapat menjalankan skrip inisialisasi instans seperti `initializeDisks.ps1` secara paralel dengan skrip `InitializeInstance.ps1`. Jika

skrip `InitializeInstance.ps1` melakukan boot ulang instans, mungkin akan mengganggu tugas terjadwal lainnya yang berjalan pada startup instans. Untuk menghindari potensi konflik, sebaiknya Anda menambahkan logika ke skrip `initializeDisks.ps1` untuk memastikan bahwa inisialisasi instans telah selesai terlebih dahulu.

> **Note**
>
> Jika skrip EC2Launch tidak menginisialisasi volume, pastikan volume tersebut online. Jika volume offline, jalankan perintah berikut untuk menjadikan semua disk online.
>
> ```
> PS C:\> Get-Disk | Where-Object IsOffline -Eq $True | Set-Disk -IsOffline
>  $False
> ```

## Lakukan boot instans Windows EC2 ke Directory Service Restore Mode (DSRM)

Jika sebuah instans yang menjalankan Microsoft Active Directory mengalami gagal sistem atau masalah kritis lainnya, Anda dapat memecahkan masalah instans dengan melakukan boot ke versi khusus Mode Aman yang disebut Directory Service Restore Mode (DSRM). Di DSRM Anda dapat memperbaiki atau memulihkan Direktori Aktif.

### Dukungan driver untuk DSRM

Cara Anda mengaktifkan DSRM dan melakukan boot ke instans bergantung pada driver yang dijalankan instans tersebut. Di konsol EC2, Anda dapat melihat detail versi driver untuk sebuah instans dari Log Sistem. Tabel berikut menunjukkan driver mana yang didukung untuk DSRM.

| Versi Driver | DSRM Didukung? | Langkah Berikutnya |
|---|---|---|
| Citrix PV 5.9 | Tidak | Pulihkan instans dari cadangan. Anda tidak dapat mengaktifkan DSRM. |
| AWS PV 7.2.0 | Tidak | Meskipun DSRM tidak didukung untuk driver ini, Anda masih dapat melepaskan volume root dari instans, mengambil snapshot volume atau membuat AMI darinya, dan memasangnya ke instans lain di Zona Ketersediaan |

| Versi Driver | DSRM Didukung? | Langkah Berikutnya |
| --- | --- | --- |
| | | yang sama dengan volume sekunder. Kemudian, Anda dapat mengaktifkan DSRM (seperti yang dijelaskan di bagian ini). |
| AWS PV 7.2.2 dan yang lebih baru | Ya | Lepaskan volume root, pasang ke instans lain, dan aktifkan DSRM (seperti yang dijelaskan di bagian ini). |
| Jaringan yang Ditingkatkan | Ya | Lepaskan volume root, pasang ke instans lain, dan aktifkan DSRM (seperti yang dijelaskan di bagian ini). |

Untuk informasi tentang cara mengaktifkan jaringan yang disempurnakan, lihatthe section called “Adaptor Jaringan Elastis (ENA)”. Untuk informasi tentang memutakhirkan AWS driver PV, lihat Upgrade driver PV pada instans Windows.

## Konfigurasikan sebuah instans untuk melakukan boot ke DSRM

Instans Windows EC2 tidak memiliki konektivitas jaringan sebelum sistem operasi berjalan. Karena alasan ini, Anda tidak dapat menekan tombol F8 pada papan tombol Anda untuk memilih opsi boot. Anda harus menggunakan salah satu dari prosedur berikut untuk melakukan boot instans Server Windows EC2 ke DSRM.

Jika Anda mencurigai bahwa Direktori Aktif telah rusak dan instans masih berjalan, Anda dapat mengonfigurasi instans tersebut untuk boot ke DSRM menggunakan kotak dialog Konfigurasi Sistem atau prompt perintah.

Untuk boot instans online ke DSRM menggunakan kotak dialog Konfigurasi Sistem

1. Di kotak dialog Jalankan, ketik `msconfig` dan tekan Enter.
2. Pilih tab Boot.
3. Di bawah Opsi boot pilih Boot aman.
4. Pilih perbaikan Direktori Aktif, lalu pilih OK. Sistem meminta Anda untuk melakukan boot ulang server.

Untuk melakukan boot instans online ke DSRM menggunakan baris perintah

Dari jendela Prompt Perintah, jalankan perintah berikut:

```
bcdedit /set safeboot dsrepair
```

Jika sebuah instans sedang offline dan tidak dapat dijangkau, Anda harus melepaskan volume root serta memasangnya ke instans lain untuk mengaktifkan mode DSRM.

Untuk melakukan boot instans offline ke DSRM

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Cari dan pilih instans yang terpengaruh. Pilih Status instans, Hentikan instans.
4. Pilih Luncurkan instans dan buat instans sementara di Zona Ketersediaan yang sama dengan instans yang terpengaruh. Pilih tipe instans yang menggunakan versi Windows yang berbeda. Misalnya, jika instance Anda adalah Windows Server 2016, maka pilih instance Windows Server 2019.

   > **Important**
   >
   > Jika Anda tidak membuat instans di Zona Ketersediaan yang sama dengan instans yang terpengaruh, Anda tidak akan dapat melampirkan volume root dari instans yang terpengaruh ke instans baru.

5. Pada panel navigasi, pilih Volume.
6. Cari volume root dari instans yang terpengaruh. Lepaskan volume dan pasang ke instans sementara yang Anda buat sebelumnya. Lampirkan dengan nama perangkat default (xvdf).
7. Gunakan Desktop Jarak Jauh untuk menyambung ke instans sementara, lalu gunakan utilitas Manajemen Disk agar volume tersedia untuk digunakan.
8. Buka prompt perintah dan jalankan perintah berikut. Ganti D dengan huruf drive sebenarnya dari volume sekunder yang baru saja Anda lampirkan:

   ```
   bcdedit /store D:\Boot\BCD /set {default} safeboot dsrepair
   ```

9. Di Utilitas Manajemen Disk, pilih drive yang Anda pasang sebelumnya, buka menu konteks (klik kanan), dan pilih Offline.

10. Di konsol EC2, lepaskan volume yang terpengaruh dari instans sementara dan pasang kembali ke instans asli Anda dengan nama perangkat `/dev/sda1`. Anda harus menentukan nama perangkat ini untuk menetapkan volume sebagai volume root.
11. Mulai instans.
12. Setelah instans lulus pemeriksaan kondisi di konsol EC2, hubungkan ke instans menggunakan Desktop Jarak Jauh, dan verifikasi bahwa instans melakukan boot ke mode DSRM.
13. (Opsional) Hapus atau hentikan instans sementara yang Anda buat dalam prosedur ini.

## Instans kehilangan konektivitas jaringan atau tugas terjadwal tidak berjalan saat diharapkan

Jika Anda memulai ulang instans dan kehilangan konektivitas jaringan, mungkin instans tersebut memiliki waktu yang salah.

Secara default, instans Windows menggunakan Waktu Universal Terkoordinasi (UTC). Jika Anda menyetel waktu pada instans ke zona waktu yang berbeda lalu memulai ulang, waktu tersebut menjadi offset dan instans kehilangan alamat IP-nya untuk sementara. Instans tersebut akhirnya mendapatkan kembali konektivitas jaringan, tetapi ini dapat memakan waktu beberapa jam. Jumlah waktu yang diperlukan instans untuk mendapatkan kembali konektivitas jaringan bergantung pada perbedaan antara UTC dan zona waktu lainnya.

Masalah waktu yang sama ini juga dapat mengakibatkan tugas terjadwal tidak berjalan seperti yang Anda harapkan. Dalam kasus ini, tugas terjadwal tidak berjalan sesuai harapan karena waktu instans salah.

Untuk menggunakan zona waktu selain UTC secara terus menerus, Anda harus menyetelRealTimeIsUniversal kunci registri. Tanpa kunci ini, instans akan menggunakan UTC setelah Anda memulai ulang.

Untuk mengatasi masalah waktu yang menyebabkan hilangnya konektivitas jaringan

1. Pastikan Anda menjalankan driver PV yang direkomendasikan. Untuk informasi selengkapnya, lihat the section called “Mutakhirkan driver PV”.
2. Verifikasi bahwa kunci registri berikut ada dan disetel ke1:  HKEY_LOCAL_MACHINE\ SYSTEM\ \ Control\\ CurrentControlSet TimeZoneInformation RealTimeIsUniversal

## Tidak bisa mendapatkan output konsol

Pada instans Windows, konsol instans menampilkan output dari tugas-tugas yang dilakukan selama proses boot Windows. Jika Windows berhasil melakukan boot, pesan terakhir yang dicatat adalah `Windows is Ready to use`. Anda juga dapat menampilkan pesan log peristiwa di konsol, tetapi fitur ini mungkin tidak diaktifkan secara default tergantung pada versi Windows Anda. Untuk informasi selengkapnya, lihat the section called “Agen peluncuran Windows”.

Untuk mendapatkan output konsol pada instans Anda menggunakan konsol Amazon EC2, pilih instans, lalu pilih Tindakan, Pantau dan atasi masalah, Dapatkan log sistem. Untuk mendapatkan output konsol menggunakan baris perintah, gunakan salah satu perintah berikut: get-console-output (AWS CLI) atau (). Get-EC2ConsoleOutput AWS Tools for Windows PowerShell

Pada instans yang menjalankan Windows Server 2012 R2 dan versi sebelumnya, jika output konsol tersebut kosong, itu bisa menunjukkan masalah dengan layanan EC2Config, seperti file konfigurasi yang salah konfigurasi, atau Windows gagal melakukan boot dengan benar. Untuk mengatasi masalah ini, unduh dan instal versi terbaru EC2Config. Untuk informasi selengkapnya, lihat the section called “Instal EC2Config”.

## Windows Server 2012 R2 tidak tersedia di jaringan

Untuk informasi tentang pemecahan masalah instans Windows Server 2012 R2 yang tidak tersedia di jaringan, lihat Windows Server 2012 R2 kehilangan konektivitas jaringan dan penyimpanan setelah reboot instance.

## Tabrakan tanda tangan disk

Anda dapat memeriksa dan menyelesaikan tabrakan tanda tangan disk menggunakan EC2Rescue untuk Windows Server. Atau, Anda dapat mengatasi masalah tanda tangan disk secara manual dengan melakukan langkah-langkah berikut.

> **Warning**
>
> Prosedur berikut menjelaskan cara mengedit Windows Registry menggunakan Registry Editor. Jika Anda tidak terbiasa dengan Windows Registry atau cara membuat perubahan dengan aman menggunakan Registry Editor, lihat Konfigurasi Registry.

1. Buka prompt perintah, ketik regedit.exe, dan tekan Enter.

2. Di Registry Editor, pilih HKEY_LOCAL_MACHINE dari menu konteks (klik kanan), lalu pilih Temukan.
3. Ketik Windows Boot Manager, lalu pilih Temukan Berikutnya.
4. Pilih kunci bernama `11000001`. Kunci ini adalah kelompok dari kunci yang Anda temukan di langkah sebelumnya.
5. Pada panel kanan, pilih `Element`, lalu pilih Ubah dari menu konteks (klik kanan).
6. Temukan tanda tangan disk empat bita pada offset 0x38 dalam data tersebut. Ini adalah tanda tangan Boot Configuration Database (BCD). Balikkan bita tersebut untuk membuat tanda tangan disk, dan tuliskan. Misalnya, tanda tangan disk yang diwakili oleh data berikut ini adalah `E9EB3AA5`:

```
...
0030  00 00 00 00 01 00 00 00
0038  A5 3A EB E9 00 00 00 00
0040  00 00 00 00 00 00 00 00
...
```

7. Di jendela Command Prompt, jalankan perintah berikut untuk memulai Microsoft DiskPart.

```
diskpart
```

8. Jalan `select disk` DiskPart kan perintah dan tentukan nomor disk untuk volume dengan tabrakan tanda tangan disk.

> **Tip**
>
> Untuk memeriksa nomor disk pada volume dengan tabrakan tanda tangan disk, gunakan utilitas Manajemen Disk. Buka prompt perintah, ketik `compmgmt.msc`, dan tekan Enter. Di panel navigasi sebelah kiri, buka Manajemen Disk. Di utilitas Manajemen Disk, periksa nomor disk untuk volume offline dengan tabrakan tanda tangan disk.

```
DISKPART> select disk 1
Disk 1 is now the selected disk.
```

9. Jalankan DiskPart perintah berikut untuk mendapatkan tanda tangan disk.

```
DISKPART>  uniqueid disk
```

```
Disk ID: 0C764FA8
```

10. Jika tanda tangan disk yang ditampilkan pada langkah sebelumnya tidak cocok dengan tanda tangan disk yang Anda tulis sebelumnya, gunakan DiskPart perintah berikut untuk mengubah tanda tangan disk sehingga cocok:

```
DISKPART> uniqueid disk id=E9EB3AA5
```

# Kernel debugging untuk instance Windows melalui jaringan

Modul Ekstensibilitas KDNET untuk Adaptor Jaringan Elastis (ENA) adalah lapisan dukungan driver perangkat keras yang memungkinkan debugging kernel Windows melalui jaringan melalui ENA pada instans Amazon Elastic Compute Cloud. Anda dapat menggunakan modul ekstensibilitas dengan Windows Debugger (WinDbg) untuk melakukan debugging tingkat kernel pada instans EC2 yang menjalankan Windows.

Kernel debugging membantu Anda mendiagnosis dan memecahkan masalah sistem operasi tingkat rendah seperti kesalahan layar biru (BSOD), kegagalan driver, dan masalah boot pada instans EC2 Windows Anda.

Daftar Isi


- Prasyarat
- Langkah 1: Instal Windows Debugging Tools pada host debug
- Langkah 2: Siapkan target debug
- Langkah 3: Mulai sesi debugging pada host debug
- Langkah 4: Reboot target debug
- Bersihkan pengaturan debug setelah debugging
- Batasan
- Catatan tambahan


## Prasyarat

Sebelum Anda mulai, pastikan Anda memiliki yang berikut:

Dua instance EC2 Windows, di subnet yang sama:

- Sebuah instance host debug — menjalankan Windows Debugger (). WinDbg
- Instance target debug — instance yang ingin Anda debug.

Untuk informasi selengkapnya tentang peluncuran instance, lihatMulai tutorial.

Grup keamanan untuk host dan instans target harus mengizinkan lalu lintas UDP masuk dan keluar pada port yang digunakan untuk debugging KDNET (kisaran yang disarankan: 50000—50039). Cara termudah untuk mengatur ini adalah dengan membuat grup keamanan dengan aturan masuk yang memungkinkan lalu lintas UDP dari dirinya sendiri sebagai sumbernya, dan kemudian melampirkan grup keamanan itu ke kedua instance. Untuk informasi selengkapnya, lihat Aturan grup keamanan di Panduan Pengguna Amazon VPC.

Instans target debug harus menjalankan salah satu versi Windows ini (atau yang lebih baru):

- Windows Server 2025 dengan nomor build 26100.7462 (patch Desember 2025)
- Windows 11 24H2 dengan nomor build 26100.7309
- Windows 11 25H2 dengan nomor build 26200.7309

> **Note**
>
> Modul ekstensibilitas KDNET untuk ENA didistribusikan sebagai bagian dari Windows dan hanya dapat diperbarui melalui pembaruan kumulatif bulanan Windows. Kami menyarankan agar Windows KB terbaru diinstal pada target debug untuk memastikan Anda memiliki versi terbaru.
> Untuk memverifikasi bahwa modul hadir pada target debug, jalankan perintah berikut dalam PowerShell sesi yang ditinggikan:
>
> ```
> Test-Path C:\Windows\system32\kd_02_1d0f.dll
> ```
>
> Jika perintah kembali`True`, modul tersedia.

## Langkah 1: Instal Windows Debugging Tools pada host debug

Instal Windows Debugging Tools pada instance host debug dengan menjalankan perintah berikut dalam satu PowerShell sesi:

```
winget install microsoft.windbg
```

Untuk petunjuk penginstalan terperinci, lihat Menginstal Debugger Windows di dokumentasi Microsoft.

Setelah instalasi, verifikasi bahwa debugger bekerja dengan menjalankan perintah berikut dalam sesi: PowerShell

```
windbgx
```

WinDbg Jendela harus terbuka. Jika ya, instalasi berhasil dan Anda dapat menutup jendela.

## Langkah 2: Siapkan target debug

> **Note**
>
> Saat debugging kernel aktif, perangkat ENA yang digunakan untuk sesi debugging didedikasikan hanya untuk lalu lintas debugger. Jika Anda perlu mempertahankan akses internet pada instance target debug selama debugging, lampirkan ENA kedua ke instance sebelum Anda mulai.

Pada target debug, buka PowerShell sesi yang ditinggikan dan konfigurasikan debugging kernel menggunakan langkah-langkah berikut.

Jalankan perintah berikut untuk mencantumkan bus, perangkat, dan nomor fungsi adaptor ENA yang terpasang pada instance:

```
Get-NetAdapter -Physical |
    Where-Object -Property PnPDeviceID -Match -Value '^PCI\\VEN_1D0F&DEV_EC2[01]&' |
    Get-NetAdapterHardwareInfo |
    Select-Object InterfaceDescription, BusNumber, DeviceNumber, FunctionNumber |
    Format-List
```

Jika beberapa adaptor ENA dilampirkan ke instance

Jika beberapa adaptor ENA dilampirkan ke instance, gunakan perintah berikut untuk memetakan setiap adaptor fisik ke alamat IP pribadinya. Anda dapat mereferensikan detail ini di Konsol Manajemen AWS bawah EC2 → Instans → [ID Instans] → Jaringan → Antarmuka jaringan. Ini

membantu Anda menghubungkan Antarmuka Jaringan tertentu IDs, pribadi IPs, dan grup keamanan dengan adaptor tingkat OS untuk debugging yang ditargetkan.

```
Get-NetAdapter -Physical |
    Where-Object PnPDeviceID -Match '^PCI\\VEN_1D0F&DEV_EC2[01]&' |
    ForEach-Object {
        $adapter = $_
        $hwInfo = $adapter | Get-NetAdapterHardwareInfo
        $ipInfo = Get-NetIPAddress -InterfaceIndex $adapter.InterfaceIndex -
AddressFamily IPv4
        [PSCustomObject]@{
            InterfaceDescription = $adapter.InterfaceDescription
            IPAddress            = $ipInfo.IPAddress
            BusNumber            = $hwInfo.BusNumber
            DeviceNumber         = $hwInfo.DeviceNumber
            FunctionNumber       = $hwInfo.FunctionNumber
        }
    } | Format-List
```

Perhatikan`BusNumber`,`DeviceNumber`, dan `FunctionNumber` nilai adaptor ENA yang akan digunakan untuk debugging dari output.

Jalankan perintah berikut untuk mengkonfigurasi debugging kernel. Ganti nilai placeholder dengan konfigurasi spesifik Anda:

```
bcdedit /debug on
bcdedit /set loadoptions FORCEHVTONOTSHAREDEBUGDEVICE
bcdedit /dbgsettings net hostip:host-private-ip port:port-number key:encryption-key
 busparams:b.d.f
```

> **Note**
>
> Periksa yang ada `loadoptions` dengan menjalankan perintah berikut. Jika nilai dikembalikan, salin string itu dan tambahkan `;FORCEHVTONOTSHAREDEBUGDEVICE` ke dalamnya.
>
> ```
> (bcdedit /enum) -match "loadoptions"
> ```

Di mana:

- *host-private-ip*— IPv4 Alamat pribadi dari instance host debug. Jika instans Anda diluncurkan di subnet IPv6 -only, lihat Menyiapkan KDNET dengan dokumentasi Microsoft untuk mempelajari lebih lanjut IPv6 tentang penggunaan dengan KDNET. IPv6
- *port-number*— Port yang akan digunakan untuk sesi debugging. Kisaran yang disarankan adalah 50000—50039 (misalnya,). `50000`
- *encryption-key*— Kunci 256-bit yang digunakan untuk mengenkripsi koneksi debugging, ditentukan sebagai empat nilai 64-bit yang dipisahkan oleh periode. Setiap nilai 64-bit dapat mencapai panjang 13 karakter hanya dengan menggunakan huruf kecil a—z dan angka 0-9. Karakter khusus tidak diperbolehkan. Contoh kunci enkripsi:`1kdnet2keys3.4kdnet5keys6.7kdnet8keys9.10kdnet11ke`.
- *b.d.f*— Bus, perangkat, dan nomor fungsi untuk perangkat ENA, diformat sebagai `bus.device.function` (misalnya,`0.3.0`), digunakan untuk debugging.

> **Tip**
>
> Untuk men-debug proses boot Windows, jalankan juga perintah berikut:
>
> ```
> bcdedit /bootdebug on
> ```

## Langkah 3: Mulai sesi debugging pada host debug

Untuk memungkinkan lalu lintas debugging pada host, Anda dapat membuat aturan firewall untuk WinDbg aplikasi atau port UDP tertentu.

Opsi 1: Izinkan WinDbg aplikasi

Jalankan perintah berikut untuk mengotorisasi WinDbg executable:

```
$WinDbgxPath = "$env:LocalAppData\Microsoft\WindowsApps\WinDbgX.exe"
New-NetFirewallRule -DisplayName "Allow Inbound KDNET Connection" -Action Allow -
Program $WinDbgxPath
```

Opsi 2: Izinkan port UDP tertentu

Atau, Anda dapat mengizinkan lalu lintas UDP masuk ke port yang dikonfigurasi untuk debugging kernel. Ganti *port-number* dengan port KDNET pilihan Anda:

```
$DebugPort = port-number
New-NetFirewallRule -DisplayName "Allow Inbound KDNET Connection" -Direction Inbound -
LocalPort $DebugPort -Protocol UDP -Action Allow
```

> **Note**
>
> Konfigurasi firewall mungkin dibatasi oleh Kebijakan Grup Domain (GPO) atau memerlukan izin Administrator yang lebih tinggi. Jika perintah gagal, hubungi administrator jaringan Anda.

Mulailah WinDbg dengan port dan kunci yang cocok dengan konfigurasi target debug. Anda dapat menentukan opsi debugging kernel tambahan yang didokumentasikan dalam referensi baris WinDbg perintah Microsoft agar sesuai dengan kasus penggunaan Anda.

```
windbgx -k net:port=port-number,key=encryption-key
```

WinDbg membuka dan menunggu target debug terhubung.

## Langkah 4: Reboot target debug

Pada target debug, restart instance untuk memulai koneksi debugging:

```
shutdown -r -t 0
```

Setelah target debug dimulai ulang, WinDbg pada host debug terhubung ke target secara otomatis. Anda sekarang dapat menggunakan WinDbg untuk memeriksa status kernel, mengatur breakpoint, dan mendiagnosis masalah.

## Bersihkan pengaturan debug setelah debugging

Pada target debug

Setelah Anda selesai debugging, hapus konfigurasi debugging kernel dari target debug untuk mengembalikan perilaku boot normal. Pada target debug, buka PowerShell sesi yang ditinggikan dan jalankan perintah berikut:

```
bcdedit /debug off
bcdedit /dbgsettings LOCAL
```

```
bcdedit /deletevalue loadoptions
```

Jika Anda memiliki `loadoptions` selain`FORCEHVTONOTSHAREDEBUGDEVICE`, Anda harus mengembalikan pengaturan dengan menjalankan `bcdedit /set loadoptions` dengan aslinya`loadoptions`.

Jika Anda mengaktifkan boot debugging, jalankan juga:

```
bcdedit /bootdebug off
```

Mulai ulang instance agar perubahan diterapkan.

Pada host debug

Hapus aturan firewall dengan menjalankan:

```
Remove-NetFirewallRule -DisplayName "Allow Inbound KDNET Connection"
```

## Batasan

Modul ekstensibilitas KDNET untuk ENA saat ini tidak mendukung:

- Jenis instans x86_64 non-logam generasi ke-8 (misalnya,) `m8a.xlarge`
- Jenis instans x86_64 non-logam 48xlarge generasi ke-7 (misalnya,) `m7a.48xlarge`
- jenis contoh u7i

Modul saat ini tidak mendukung instance dengan Boot Aman diaktifkan. Anda dapat memverifikasi status dengan menjalankan `Confirm-SecureBootUEFI` dalam PowerShell sesi yang ditinggikan. Jika outputnya`True`, Secure Boot aktif. Perhatikan bahwa semua gambar yang disediakan Amazon dengan awalan 'TPM' mengaktifkan Boot Aman secara default.

## Catatan tambahan

Jika Anda mengalami masalah saat menghubungkan debugger ke instance target, verifikasi hal berikut:

- Semua prasyarat yang tercantum dalam panduan ini terpenuhi, termasuk versi build Windows Server yang diperlukan dan keberadaan modul ekstensibilitas.

- Grup keamanan yang dilampirkan ke kedua instance dikonfigurasi dengan benar untuk memungkinkan lalu lintas di antara mereka pada port debugging yang dikonfigurasi.
- Aturan Windows Firewall pada instance host tidak memblokir lalu lintas jaringan antara dua instance pada port yang dikonfigurasi.

Untuk panduan tambahan, lihat Mengatur debug kernel jaringan KDNET secara manual di dokumentasi Microsoft.

# Setel ulang kata sandi administrator Windows untuk instans Windows Amazon EC2

Jika Anda tidak lagi dapat terhubung ke instans Windows Amazon EC2 Anda karena kata sandi administrator Windows hilang atau kedaluwarsa, Anda dapat mengatur ulang kata sandi.

> **Note**
>
> Ada dokumen AWS Systems Manager Otomasi yang secara otomatis menerapkan langkah-langkah manual yang diperlukan untuk mengatur ulang kata sandi administrator lokal. Untuk informasi selengkapnya, lihat Mengatur ulang kata sandi dan kunci SSH pada instans EC2 di Panduan Pengguna.AWS Systems Manager

Metode manual untuk mengatur ulang kata sandi administrator EC2Launch v2, EC2Config, atau EC2Launch.

- Untuk semua AMI Windows yang didukung dan menyertakan agen EC2Launch v2, gunakan EC2Launch v2.
- Untuk AMI Windows sebelum Windows Server 2016, gunakan layanan EC2Config.
- Untuk Windows Server 2016 dan versi AMI yang lebih baru, gunakan layanan EC2Launch.

Prosedur ini juga menjelaskan cara terhubung ke instans jika Anda kehilangan pasangan kunci yang digunakan untuk membuat instans tersebut. Amazon EC2 menggunakan kunci publik untuk mengenkripsi sebuah data, seperti kata sandi, dan kunci privat untuk mendekripsi data. Kunci publik dan privat dikenal sebagai pasangan kunci. Dengan instans Windows, Anda dapat menggunakan pasangan kunci untuk mendapatkan kata sandi administrator, lalu masuk menggunakan RDP.

> **Note**
>
> Jika Anda telah menonaktifkan akun administrator lokal pada instans dan instans Anda dikonfigurasi untuk Systems Manager, Anda juga dapat mengaktifkan ulang dan mengatur ulang kata sandi administrator lokal menggunakan EC2Rescue dan Run Command. Untuk informasi selengkapnya, lihat Menggunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command.

Daftar Isi


- Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2launch v2
- Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2launch
- Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2config


## Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2launch v2

Jika Anda kehilangan kata sandi administrator Windows dan menggunakan AMI Windows yang didukung, serta menyertakan agen EC2Launch v2, Anda dapat menggunakan EC2Launch v2 untuk membuat kata sandi baru.

Jika Anda menggunakan Windows Server 2016 atau AMI versi yang lebih baru dan tidak menyertakan agen EC2Launch v2, lihat Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2launch.

Jika Anda menggunakan AMI Windows Server atau versi yang lebih baru dari Windows Server 2016 dan tidak menyertakan agen EC2Launch v2, lihat. Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2config

> **Note**
>
> Jika Anda telah menonaktifkan akun administrator lokal pada instans dan instans Anda dikonfigurasi untuk Systems Manager, Anda juga dapat mengaktifkan ulang dan mengatur ulang kata sandi administrator lokal menggunakan EC2Rescue dan Run Command. Untuk informasi selengkapnya, lihat Menggunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command.

> **Note**
>
> Ada dokumen AWS Systems Manager Otomasi yang secara otomatis menerapkan langkah-langkah manual yang diperlukan untuk mengatur ulang kata sandi administrator lokal. Untuk informasi selengkapnya, lihat Mengatur ulang kata sandi dan kunci SSH pada instans EC2 di Panduan Pengguna.AWS Systems Manager

Untuk mengatur ulang kata sandi administrator Windows menggunakan EC2Launch v2, Anda harus melakukan hal berikut ini:

- Langkah 1: Verifikasi bahwa agen EC2Launch v2 sedang berjalan
- Langkah 2: Copot volume root dari instans
- Langkah 3: Lampirkan volume ke instans sementara
- Langkah 4: Hapus .run-sekali file
- Langkah 5: Mulai ulang instans asli

## Langkah 1: Verifikasi bahwa agen EC2Launch v2 sedang berjalan

Sebelum Anda mencoba untuk mengatur ulang kata sandi administrator, verifikasi bahwa agen EC2Launch v2 terinstal dan berjalan. Anda menggunakan agen EC2Launch v2 untuk mengatur ulang kata sandi administrator nanti di bagian ini.

Untuk memverifikasi bahwa agen EC2Launch v2 sedang berjalan

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/
2. Di panel navigasi, pilih Instans, lalu pilih instans yang membutuhkan pengaturan ulang kata sandi. Instans ini dirujuk sebagai instans asli di dalam prosedur ini.
3. Pilih Tindakan, Pantau dan pecahkan masalah, Dapatkan log sistem .
4. Temukan entri Luncurkan EC2, misalnya Luncurkan: EC2Launch v2 service v2.0.124. Jika Anda melihat entri ini, berarti layanan EC2Launch v2 sedang berjalan.

   Jika output log sistem kosong, atau jika agen EC2Launch v2 tidak berjalan, pecahkan masalah instans menggunakan layanan Instance Console Screenshot. Untuk informasi selengkapnya, lihat Mengambil tangkapan layar instans yang tidak dapat dijangkau.

## Langkah 2: Copot volume root dari instans

Anda tidak dapat menggunakan EC2Launch v2 untuk mengatur ulang kata sandi administrator, jika volume tempat kata sandi disimpan dan dilampirkan ke instans sebagai volume root. Anda harus mencopot volume dari instans asli sebelum dapat melampirkannya ke instans sementara sebagai volume sekunder.

Untuk mencopot volume root dari instans

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/
2. Di panel navigasi, pilih Instans.
3. Pilih instance yang memerlukan pengaturan ulang kata sandi dan pilih Status instans, Stop instance. Setelah status instans berubah menjadi Berhenti, lanjutkan dengan langkah berikutnya.
4. (Opsional) Jika Anda memiliki kunci privat yang Anda tentukan saat meluncurkan instans ini, lanjutkan dengan langkah berikutnya. Jika tidak, gunakan langkah-langkah berikut untuk mengganti instans dengan instans baru yang Anda luncurkan dengan sebuah pasangan kunci baru.

    a. Buatlah pasangan kunci baru menggunakan konsol Amazon EC2. Untuk memberikan nama pasangan kunci baru Anda sama seperti nama kunci privat yang hilang, Anda harus menghapus pasangan kunci yang sudah ada terlebih dahulu.

    b. Pilih instans yang ingin diganti. Catat tipe instans, VPC, subnet, grup keamanan, dan peran IAM dari instans.

    c. Dengan instance yang dipilih, pilih Actions, Image and templates, Create image. Ketikkan nama dan deskripsi untuk gambar dan pilih Buat gambar.

    d. Di panel navigasi, pilih AMI. Tunggu status gambar berubah menjadi tersedia. Kemudian, pilih gambar dan pilih Launch instance dari AMI.

    e. Lengkapi bidang untuk meluncurkan instance, pastikan untuk memilih jenis instance yang sama, VPC, subnet, grup keamanan, dan peran IAM sebagai instance yang akan diganti, lalu pilih Launch instance.

    f. Saat diminta, pilih key pair yang Anda buat untuk instance baru, lalu pilih Launch instance.

    g. (Opsional) Jika instans asli memiliki alamat IP Elastis terkait, alihkan ke instans baru. Jika instans asli memiliki volume EBS selain volume root, alihkan ke instans baru.

5. Copot volume root dari instans asli dengan cara sebagai berikut:

a. Pilih instance asli dan pilih tab Penyimpanan. Perhatikan nama perangkat root di bawah nama perangkat Root. Temukan volume dengan nama perangkat ini di bawah Blokir perangkat, dan catat ID volume.

b. Pada panel navigasi, pilih Volume.

c. Dalam daftar volume, pilih volume yang Anda catat sebagai perangkat root, dan pilih Tindakan, Lepaskan Volume. Setelah status volume berubah menjadi tersedia, lanjutkan ke langkah berikutnya.

6. Jika Anda membuat instance baru untuk menggantikan instance asli Anda, Anda dapat menghentikan instance asli sekarang. Ini tidak lagi dibutuhkan. Untuk sisa prosedur ini, semua referensi ke instance asli berlaku untuk instance baru yang Anda buat.

## Langkah 3: Lampirkan volume ke instans sementara

Selanjutnya, luncurkan instans sementara dan lampirkan volume ke instans tersebut sebagai volume sekunder. Ini adalah instans yang Anda gunakan untuk mengubah file konfigurasi.

Untuk meluncurkan sebuah instans sementara dan melampirkan volume

1. Luncurkan instans sementara dengan cara sebagai berikut:

a. Di panel navigasi, pilih Instans, pilih Luncurkan instans, lalu pilih AMI.

> **Important**
>
> Untuk menghindari tabrakan tanda tangan disk, Anda harus memilih AMI untuk versi Windows yang berbeda. Misalnya, jika instans asli menjalankan Windows Server 2019, luncurkan instans sementara menggunakan AMI dasar untuk Windows Server 2016.

b. Abaikan tipe instans default dan pilih Berikutnya: Konfigurasi Detail Instans.

c. Pada halaman Konfigurasi Detail Instans, untuk Subnet, pilih Zona Ketersediaan yang sama dengan instans asli dan pilih Tinjau dan Luncurkan.

> **Important**
>
> Instans sementara harus berada dalam Zona Ketersediaan yang sama dengan instans asli. Jika instans sementara Anda berada dalam Zona Ketersediaan yang berbeda, Anda tidak dapat melampirkan volume root instans asli ke instans tersebut.

d. Di halaman Tinjau Peluncuran Instans, pilih Luncurkan.

e. Jika diminta, buat pasangan kunci baru, unduh ke lokasi yang aman di komputer Anda, lalu pilih Luncurkan Instans.

2. Lampirkan volume ke instans sementara sebagai volume sekunder dengan cara sebagai berikut:

a. Di panel navigasi, pilih Volume, pilih volume yang ingin Anda copot dari instans asli, lalu pilih Tindakan, Lampirkan Volume.

b. Dalam kotak dialog Lampirkan Volume, untuk Instans, mulai untuk mengetikkan nama atau ID dari instans sementara Anda, lalu pilih instans dari daftar.

c. Untuk Perangkat, ketikkan **`xvdf`** (jika belum ada di sana), lalu pilih Lampirkan.

## Langkah 4: Hapus .run-sekali file

Anda sekarang harus menghapus file `.run-once` dari volume offline yang dilampirkan ke instans. Ini mengarahkan EC2Launch v2 untuk menjalankan semua tugas dengan frekuensi `once`, yang mencakup pengaturan kata sandi administrator. Jalur file di volume sekunder yang Anda lampirkan akan mirip dengan `D:\ProgramData\Amazon\EC2Launch\state\.run-once`.

Untuk menghapus file .run-once

1. Buka utilitas Manajemen Disk, dan bawa drive online menggunakan petunjuk ini: Buat volume Amazon EBS tersedia untuk digunakan.
2. Temukan lokasi file `.run-once` di disk yang Anda bawa online.
3. Hapus file .run-once.

> **Important**
>
> Skrip apa pun yang diatur untuk dijalankan sekali akan dipicu oleh tindakan ini.

### Langkah 5: Mulai ulang instans asli

Setelah Anda menghapus file `.run-once`, lampirkan ulang volume ke instans asli sebagai volume root dan hubungkan ke instans menggunakan pasangan kuncinya untuk mengambil kata sandi administrator.

1. Lampirkan kembali volume ke instans asli dengan cara sebagai berikut:
   a. Di panel navigasi, pilih Volume, pilih volume yang ingin Anda copot dari instans sementara, lalu pilih Tindakan, Lampirkan Volume.
   b. Dalam kotak dialog Lampirkan Volume, untuk Instans, mulai untuk mengetikkan nama atau ID dari instans asli Anda, lalu pilih instans.
   c. Untuk Perangkat, ketikkan **`/dev/sda1`**.
   d. Pilih Lampirkan. Setelah status volume berubah menjadi `in-use`, lanjutkan ke langkah berikutnya.
2. Di panel navigasi, pilih Instans. Pilih instans asli dan pilih Status instans, Mulai instans. Setelah status instans berubah menjadi `Running`, lanjutkan ke langkah berikutnya.
3. Ambil kata sandi administrator Windows baru Anda menggunakan kunci privat untuk pasangan kunci baru dan hubungkan ke instans. Untuk informasi selengkapnya, lihat Hubungkan ke instans Windows Anda menggunakan RDP.

   > **Important**
   >
   > Instans tersebut mendapatkan alamat IP publik baru setelah Anda menghentikan dan memulainya. Pastikan untuk terhubung ke instans menggunakan nama DNS publiknya saat ini. Untuk informasi selengkapnya, lihat Perubahan status instans Amazon EC2.

4. (Opsional) Jika Anda tidak menggunakan instans sementara lagi, Anda dapat mengakhirinya. Pilih instans sementara, dan pilih Status instans, Akhiri instans.

## Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2launch

Jika Anda kehilangan kata sandi administrator Windows dan menggunakan Windows Server 2016 atau versi AMI yang lebih baru, Anda dapat menggunakan alat EC2Rescue, yang menggunakan layanan EC2Launch untuk membuat kata sandi baru.

Jika Anda menggunakan Windows Server 2016 atau AMI versi yang lebih baru dan tidak menyertakan agen EC2Launch v2, Anda dapat menggunakan EC2Launch v2 untuk membuat kata sandi baru.

Jika Anda menggunakan AMI Windows Server versi sebelumnya daripada Windows Server 2016, lihat Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2config.

**Warning**

Ketika Anda menghentikan sebuah instance, data pada volume penyimpanan instance akan hilang. Untuk menyimpan data ini, buat cadangan ke penyimpanan persisten.

**Note**

Jika Anda telah menonaktifkan akun administrator lokal pada instans dan instans Anda dikonfigurasi untuk Systems Manager, Anda juga dapat mengaktifkan ulang dan mengatur ulang kata sandi administrator lokal menggunakan EC2Rescue dan Run Command. Untuk informasi selengkapnya, lihat Menggunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command.

**Note**

Ada dokumen AWS Systems Manager Otomasi yang secara otomatis menerapkan langkah-langkah manual yang diperlukan untuk mengatur ulang kata sandi administrator lokal. Untuk informasi selengkapnya, lihat Mengatur ulang kata sandi dan kunci SSH pada instans EC2 di Panduan Pengguna.AWS Systems Manager

Untuk mengatur ulang kata sandi administrator Windows menggunakan EC2Launch, Anda harus melakukan hal berikut ini:

- Langkah 1: Copot volume root dari instans
- Langkah 2: Lampirkan volume ke instans sementara
- Langkah 3: Atur ulang kata sandi administrator
- Langkah 4: Mulai ulang instans asli

## Langkah 1: Copot volume root dari instans

Anda tidak dapat menggunakan EC2Launch untuk mengatur ulang kata sandi administrator, jika volume tempat kata sandi disimpan dan dilampirkan ke instans sebagai volume root. Anda harus mencopot volume dari instans asli sebelum dapat melampirkannya ke instans sementara sebagai volume sekunder.

Untuk mencopot volume root dari instans

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/
2. Di panel navigasi, pilih Instans.
3. Pilih instance yang memerlukan pengaturan ulang kata sandi dan pilih Status instans, Stop instance. Setelah status instans berubah menjadi Berhenti, lanjutkan dengan langkah berikutnya.
4. (Opsional) Jika Anda memiliki kunci privat yang Anda tentukan saat meluncurkan instans ini, lanjutkan dengan langkah berikutnya. Jika tidak, gunakan langkah-langkah berikut untuk mengganti instans dengan instans baru yang Anda luncurkan dengan sebuah pasangan kunci baru.

    a. Buatlah pasangan kunci baru menggunakan konsol Amazon EC2. Untuk memberikan nama pasangan kunci baru Anda sama seperti nama kunci privat yang hilang, Anda harus menghapus pasangan kunci yang sudah ada terlebih dahulu.

    b. Pilih instans yang ingin diganti. Catat tipe instans, VPC, subnet, grup keamanan, dan peran IAM dari instans.

    c. Dengan instance yang dipilih, pilih Actions, Image and templates, Create image. Ketikkan nama dan deskripsi untuk gambar dan pilih Buat gambar.

    d. Di panel navigasi, pilih AMI. Tunggu status gambar berubah menjadi tersedia. Kemudian, pilih gambar dan pilih Launch instance dari AMI.

    e. Lengkapi bidang untuk meluncurkan instance, pastikan untuk memilih jenis instance yang sama, VPC, subnet, grup keamanan, dan peran IAM sebagai instance yang akan diganti, lalu pilih Launch instance.

    f. Saat diminta, pilih key pair yang Anda buat untuk instance baru, lalu pilih Launch instance.

    g. (Opsional) Jika instans asli memiliki alamat IP Elastis terkait, alihkan ke instans baru. Jika instans asli memiliki volume EBS selain volume root, alihkan ke instans baru.

5. Copot volume root dari instans asli dengan cara sebagai berikut:

a. Pilih instance asli dan pilih tab Penyimpanan. Perhatikan nama perangkat root di bawah nama perangkat Root. Temukan volume dengan nama perangkat ini di bawah Blokir perangkat, dan catat ID volume.

b. Pada panel navigasi, pilih Volume.

c. Dalam daftar volume, pilih volume yang Anda catat sebagai perangkat root, dan pilih Tindakan, Lepaskan Volume. Setelah status volume berubah menjadi tersedia, lanjutkan ke langkah berikutnya.

6. Jika Anda membuat instance baru untuk menggantikan instance asli Anda, Anda dapat menghentikan instance asli sekarang. Ini tidak lagi dibutuhkan. Untuk sisa prosedur ini, semua referensi ke instance asli berlaku untuk instance baru yang Anda buat.

## Langkah 2: Lampirkan volume ke instans sementara

Selanjutnya, luncurkan instans sementara dan lampirkan volume ke instans tersebut sebagai volume sekunder. Ini adalah instans yang Anda gunakan untuk menjalankan EC2Launch.

Untuk meluncurkan sebuah instans sementara dan melampirkan volume

1. Luncurkan instans sementara dengan cara sebagai berikut:

a. Di panel navigasi, pilih Instans, pilih Luncurkan instans, lalu pilih AMI.

> **Important**
>
> Untuk menghindari tabrakan tanda tangan disk, Anda harus memilih AMI untuk versi Windows yang berbeda. Misalnya, jika instans asli menjalankan Windows Server 2019, luncurkan instans sementara menggunakan AMI dasar untuk Windows Server 2016.

b. Abaikan tipe instans default dan pilih Berikutnya: Konfigurasi Detail Instans.

c. Pada halaman Konfigurasi Detail Instans, untuk Subnet, pilih Zona Ketersediaan yang sama dengan instans asli dan pilih Tinjau dan Luncurkan.

> **Important**
>
> Instans sementara harus berada dalam Zona Ketersediaan yang sama dengan instans asli. Jika instans sementara Anda berada dalam Zona Ketersediaan yang berbeda, Anda tidak dapat melampirkan volume root instans asli ke instans tersebut.

d. Di halaman Tinjau Peluncuran Instans, pilih Luncurkan.

e. Jika diminta, buat pasangan kunci baru, unduh ke lokasi yang aman di komputer Anda, lalu pilih Luncurkan Instans.

2. Lampirkan volume ke instans sementara sebagai volume sekunder dengan cara sebagai berikut:

a. Di panel navigasi, pilih Volume, pilih volume yang ingin Anda copot dari instans asli, lalu pilih Tindakan, Lampirkan Volume.

b. Dalam kotak dialog Lampirkan Volume, untuk Instans, mulai untuk mengetikkan nama atau ID dari instans sementara Anda, lalu pilih instans dari daftar.

c. Untuk Perangkat, ketikkan **`xvdf`** (jika belum ada di sana), lalu pilih Lampirkan.

## Langkah 3: Atur ulang kata sandi administrator

Selanjutnya, hubungkan ke instans sementara dan gunakan EC2Launch untuk mengatur ulang kata sandi administrator.

Untuk mengatur ulang kata sandi administrator

1. Hubungkan ke instans sementara dan gunakan alat EC2Rescue for Windows Server pada instans guna mengatur ulang kata sandi administrator seperti berikut ini:

a. Unduh file zip EC2Rescue untuk Windows Server, ekstrak isinya, dan jalankan. EC2Rescue.exe

b. Pada layar Perjanjian Lisensi, baca perjanjian lisensi, dan jika Anda menerima persyaratan, pilih Saya setuju.

c. Pada layar Selamat Datang di EC2Rescue for Windows Server pilih Berikutnya.

d. Pada layar Pilih mode, pilih Instans offline.

e. Pada layar Pilih disk, pilih perangkat xvdf, lalu pilih Berikutnya.

f. Konfirmasi pilihan disk dan pilih Ya.

g. Setelah volume dimuat, pilih OKE.

h. Pada layar Pilih Opsi Instans Offline, pilih Diagnosis dan Penyelamatan.

i. Pada layar Ringkasan, tinjau informasi dan pilih Berikutnya.

j. Pada layar Kemungkinan masalah yang Terdeteksi, pilih Atur Ulang Kata Sandi Administrator, lalu pilih Berikutnya.

k. Pada layar Konfirmasi, pilih Selamatkan, OKE.

l. Pada layar Selesai, pilih Akir.

m. Tutup alat EC2Rescue for Windows Server, putuskan sambungan dari instans sementara, lalu kembali ke konsol Amazon EC2.

2. Copot volume (`xvdf`) sekunder dari instans asli seperti berikut ini:

a. Pada panel navigasi, pilih Instans dan pilih instans sementara.

b. Pada tab Penyimpanan untuk instans sementara, catat ID dari volume EBS yang terdaftar sebagai xvdf.

c. Pada panel navigasi, pilih Volume.

d. Dalam daftar volume, pilih volume yang dicatat di langkah sebelumnya, lalu pilih Tindakan, Copot Volume. Setelah status volume berubah menjadi tersedia, lanjutkan ke langkah berikutnya.

## Langkah 4: Mulai ulang instans asli

Setelah Anda mengatur ulang kata sandi administrator menggunakan EC2Launch, lampirkan ulang volume ke instans asli sebagai volume root dan hubungkan ke instans menggunakan pasangan kuncinya untuk mengambil kata sandi administrator.

Untuk memulai ulang instans asli

1. Lampirkan kembali volume ke instans asli dengan cara sebagai berikut:

a. Di panel navigasi, pilih Volume, pilih volume yang ingin Anda copot dari instans sementara, lalu pilih Tindakan, Lampirkan Volume.

b. Dalam kotak dialog Lampirkan Volume, untuk Instans, mulai untuk mengetikkan nama atau ID dari instans asli Anda, lalu pilih instans.

c. Untuk Perangkat, ketikkan **`/dev/sda1`**.

d. Pilih Lampirkan. Setelah status volume berubah menjadi `in-use`, lanjutkan ke langkah berikutnya.

2. Di panel navigasi, pilih Instans. Pilih instans asli dan pilih Status instans, Mulai instans. Setelah status instans berubah menjadi `Running`, lanjutkan ke langkah berikutnya.
3. Ambil kata sandi administrator Windows baru Anda menggunakan kunci privat untuk pasangan kunci baru dan hubungkan ke instans. Untuk informasi selengkapnya, lihat Hubungkan ke instans Windows Anda menggunakan RDP.
4. (Opsional) Jika Anda tidak menggunakan instans sementara lagi, Anda dapat mengakhirinya. Pilih instans sementara, dan pilih Status instans, Akhiri instans.

## Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2config

Jika Anda kehilangan kata sandi administrator Windows dan menggunakan AMI Windows sebelum Windows Server 2016, Anda dapat menggunakan agen EC2Config untuk membuat kata sandi baru.

Jika Anda menggunakan Windows Server 2016 atau AMI yang lebih baru, lihat Setel ulang kata sandi admin Windows untuk instans EC2 menggunakan EC2launch atau, Anda dapat menggunakan alat EC2Rescue, yang menggunakan layanan EC2Launch untuk membuat kata sandi baru.

> **Note**
>
> Jika Anda telah menonaktifkan akun administrator lokal pada instans dan instans Anda dikonfigurasi untuk Systems Manager, Anda juga dapat mengaktifkan ulang dan mengatur ulang kata sandi administrator lokal menggunakan EC2Rescue dan Run Command. Untuk informasi selengkapnya, lihat Menggunakan EC2Rescue untuk Windows Server dengan Systems Manager Run Command.

> **Note**
>
> Ada dokumen AWS Systems Manager Otomasi yang secara otomatis menerapkan langkah-langkah manual yang diperlukan untuk mengatur ulang kata sandi administrator lokal. Untuk informasi selengkapnya, lihat Mengatur ulang kata sandi dan kunci SSH pada instans EC2 di Panduan Pengguna.AWS Systems Manager

Untuk mengatur ulang kata sandi administrator Windows menggunakan EC2Config, Anda harus melakukan hal berikut ini:

- Langkah 1: Verifikasi bahwa layanan EC2Config sedang berjalan
- Langkah 2: Lepaskan volume root dari instans
- Langkah 3: Lampirkan volume ke instans sementara
- Langkah 4: Ubah file konfigurasi
- Langkah 5: Mulai ulang instans asli

## Langkah 1: Verifikasi bahwa layanan EC2Config sedang berjalan

Sebelum Anda mencoba untuk mengatur ulang kata sandi administrator, verifikasi bahwa layanan EC2Config terinstal dan berjalan. Anda menggunakan layanan EC2Config untuk mengatur ulang kata sandi administrator nanti di bagian ini.

Untuk memverifikasi bahwa layanan EC2Config sedang berjalan

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/
2. Di panel navigasi, pilih Instans, lalu pilih instans yang membutuhkan pengaturan ulang kata sandi. Instans ini dirujuk sebagai instans asli di dalam prosedur ini.
3. Pilih Actions, Monitor and troubleshoot, Get system log.
4. Temukan entri EC2 Agent, misalnya EC2 Agent: Ec2Config service v3.18.1118. Jika Anda melihat entri ini, berarti layanan EC2Config sedang berjalan.

   Jika output log sistem kosong, atau jika layanan EC2Config tidak berjalan, pecahkan masalah instans menggunakan layanan Instance Console Screenshot. Untuk informasi selengkapnya, lihat Mengambil tangkapan layar instans yang tidak dapat dijangkau.

## Langkah 2: Lepaskan volume root dari instans

Anda tidak dapat menggunakan EC2Config untuk mengatur ulang kata sandi administrator, jika volume tempat kata sandi disimpan dan dilampirkan ke instans sebagai volume root. Anda harus mencopot volume dari instans asli sebelum dapat melampirkannya ke instans sementara sebagai volume sekunder.

Untuk mencopot volume root dari instans

1. Buka konsol Amazon EC2 di. https://console.aws.amazon.com/ec2/
2. Di panel navigasi, pilih Instans.
3. Pilih instance yang memerlukan pengaturan ulang kata sandi dan pilih Status instans, Stop instance. Setelah status instans berubah menjadi Berhenti, lanjutkan dengan langkah berikutnya.
4. (Opsional) Jika Anda memiliki kunci privat yang Anda tentukan saat meluncurkan instans ini, lanjutkan dengan langkah berikutnya. Jika tidak, gunakan langkah-langkah berikut untuk mengganti instans dengan instans baru yang Anda luncurkan dengan sebuah pasangan kunci baru.

   a. Buatlah pasangan kunci baru menggunakan konsol Amazon EC2. Untuk memberikan nama pasangan kunci baru Anda sama seperti nama kunci privat yang hilang, Anda harus menghapus pasangan kunci yang sudah ada terlebih dahulu.

   b. Pilih instans yang ingin diganti. Catat tipe instans, VPC, subnet, grup keamanan, dan peran IAM dari instans.

   c. Dengan instance yang dipilih, pilih Actions, Image and templates, Create image. Ketikkan nama dan deskripsi untuk gambar dan pilih Buat gambar.

   d. Di panel navigasi, pilih AMI. Tunggu status gambar berubah menjadi tersedia. Kemudian, pilih gambar dan pilih Launch instance dari AMI.

   e. Lengkapi bidang untuk meluncurkan instance, pastikan untuk memilih jenis instance yang sama, VPC, subnet, grup keamanan, dan peran IAM sebagai instance yang akan diganti, lalu pilih Launch instance.

   f. Saat diminta, pilih key pair yang Anda buat untuk instance baru, lalu pilih Launch instance.

   g. (Opsional) Jika instans asli memiliki alamat IP Elastis terkait, alihkan ke instans baru. Jika instans asli memiliki volume EBS selain volume root, alihkan ke instans baru.

5. Copot volume root dari instans asli dengan cara sebagai berikut:

   a. Pilih instance asli dan pilih tab Penyimpanan. Perhatikan nama perangkat root di bawah nama perangkat Root. Temukan volume dengan nama perangkat ini di bawah Blokir perangkat, dan catat ID volume.

   b. Pada panel navigasi, pilih Volume.

   c. Dalam daftar volume, pilih volume yang Anda catat sebagai perangkat root, dan pilih Tindakan, Lepaskan Volume. Setelah status volume berubah menjadi tersedia, lanjutkan ke langkah berikutnya.

6. Jika Anda membuat instance baru untuk menggantikan instance asli Anda, Anda dapat menghentikan instance asli sekarang. Ini tidak lagi dibutuhkan. Untuk sisa prosedur ini, semua referensi ke instance asli berlaku untuk instance baru yang Anda buat.

## Langkah 3: Lampirkan volume ke instans sementara

Selanjutnya, luncurkan instans sementara dan lampirkan volume ke instans tersebut sebagai volume sekunder. Ini adalah instans yang Anda gunakan untuk mengubah file konfigurasi.

Untuk meluncurkan sebuah instans sementara dan melampirkan volume

1. Luncurkan instans sementara dengan cara sebagai berikut:

   a. Di panel navigasi, pilih Instans, pilih Luncurkan instans, lalu pilih AMI.

   > **Important**
   >
   > Untuk menghindari tabrakan tanda tangan disk, Anda harus memilih AMI untuk versi Windows yang berbeda. Misalnya, jika instans asli menjalankan Windows Server 2019, luncurkan instans sementara menggunakan AMI dasar untuk Windows Server 2016.

   b. Abaikan tipe instans default dan pilih Berikutnya: Konfigurasi Detail Instans.

   c. Pada halaman Konfigurasi Detail Instans, untuk Subnet, pilih Zona Ketersediaan yang sama dengan instans asli dan pilih Tinjau dan Luncurkan.

   > **Important**
   >
   > Instans sementara harus berada dalam Zona Ketersediaan yang sama dengan instans asli. Jika instans sementara Anda berada dalam Zona Ketersediaan yang berbeda, Anda tidak dapat melampirkan volume root instans asli ke instans tersebut.

   d. Di halaman Tinjau Peluncuran Instans, pilih Luncurkan.

   e. Jika diminta, buat pasangan kunci baru, unduh ke lokasi yang aman di komputer Anda, lalu pilih Luncurkan Instans.

2. Lampirkan volume ke instans sementara sebagai volume sekunder dengan cara sebagai berikut:

a. Di panel navigasi, pilih Volume, pilih volume yang ingin Anda copot dari instans asli, lalu pilih Tindakan, Lampirkan Volume.

b. Dalam kotak dialog Lampirkan Volume, untuk Instans, mulai untuk mengetikkan nama atau ID dari instans sementara Anda, lalu pilih instans dari daftar.

c. Untuk Perangkat, ketikkan **`xvdf`** (jika belum ada di sana), lalu pilih Lampirkan.

## Langkah 4: Ubah file konfigurasi

Setelah Anda melampirkan volume ke instans sementara sebagai volume sekunder, ubah plugin `Ec2SetPassword` di file konfigurasi.

Untuk mengubah file konfigurasi

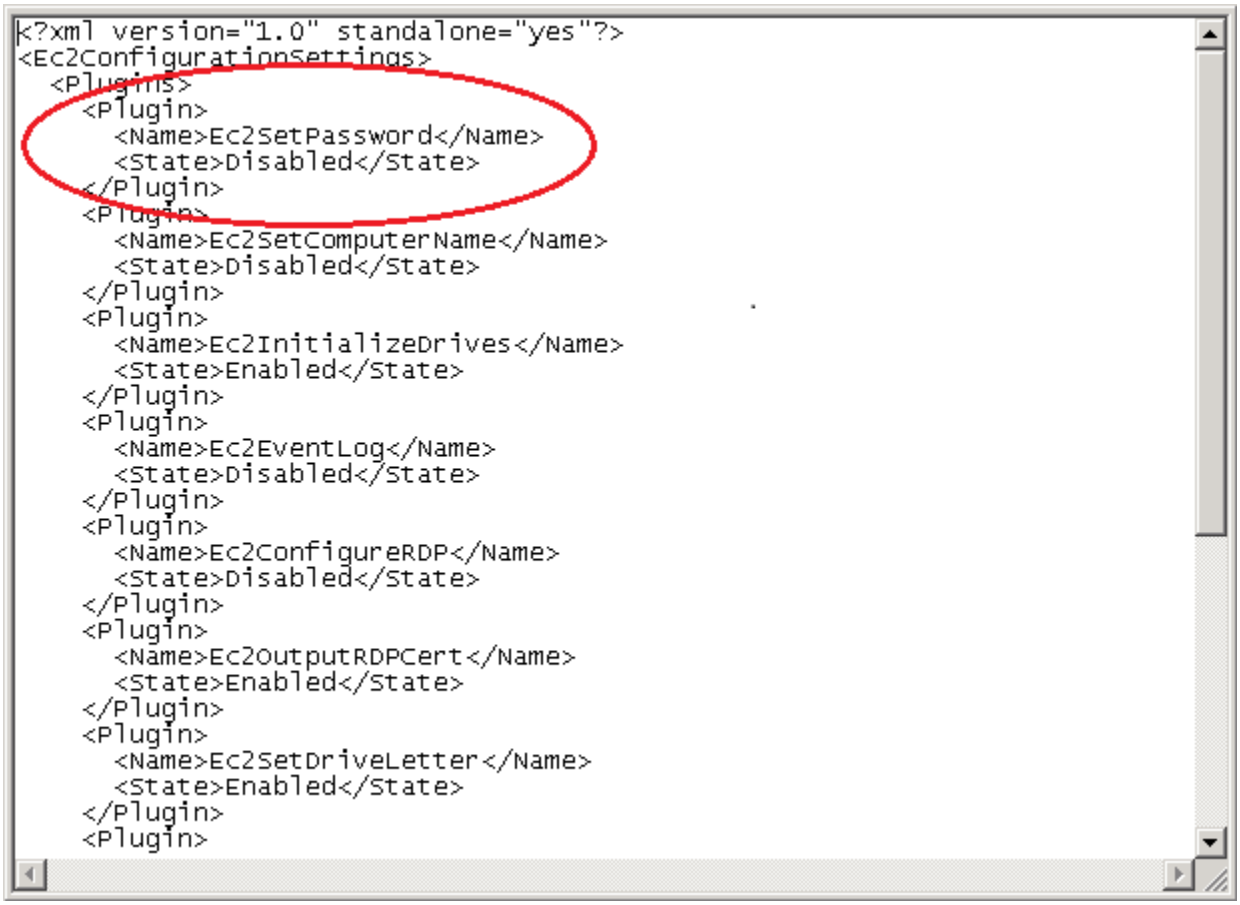
1. Dari instans sementara, ubah file konfigurasi pada volume sekunder dengan cara seperti berikut:

   a. Luncurkan dan hubungkan ke instans sementara.

   b. Gunakan petunjuk berikut untuk membawa drive online: Buat volume Amazon EBS tersedia untuk digunakan.

   c. Lakukan navigasi pada volume sekunder, dan buka `\Program Files\Amazon \Ec2ConfigService\Settings\config.xml` menggunakan editor teks, seperti Notepad.

   d. Pada bagian atas file, temukan plugin dengan nama `Ec2SetPassword`, seperti yang ditunjukkan dalam screenshot. Ubah status dari `Disabled` ke `Enabled`, lalu simpan file.




2. Setelah mengubah file konfigurasi, copot volume sekunder dari instans sementara dengan cara seperti berikut:

   a. Menggunakan utitlitas Manajemen Disk, yang membuat volume menjadi offline.

   b. Putuskan sambungan dari instans sementara lalu kembali ke konsol Amazon EC2.

   c. Dalam panel navigasi, pilih Volume, pilih volume, lalu pilih Tindakan, Copot Volume. Setelah status volume berubah menjadi tersedia, lanjutkan dengan langkah berikutnya.

## Langkah 5: Mulai ulang instans asli

Setelah Anda mengubah file konfigurasi, lampirkan ulang volume ke instans asli sebagai volume root dan hubungkan ke instans menggunakan pasangan kuncinya untuk mengambil kata sandi administrator.

1. Lampirkan kembali volume ke instans asli dengan cara sebagai berikut:

   a. Di panel navigasi, pilih Volume, pilih volume yang ingin Anda copot dari instans sementara, lalu pilih Tindakan, Lampirkan Volume.

b. Dalam kotak dialog Lampirkan Volume, untuk Instans, mulai untuk mengetikkan nama atau ID dari instans asli Anda, lalu pilih instans.

c. Untuk Perangkat, ketikkan **`/dev/sda1`**.

d. Pilih Lampirkan. Setelah status volume berubah menjadi `in-use`, lanjutkan ke langkah berikutnya.

2. Di panel navigasi, pilih Instans. Pilih instans asli dan pilih Status instans, Mulai instans. Setelah status instans berubah menjadi `Running`, lanjutkan ke langkah berikutnya.

3. Ambil kata sandi administrator Windows baru Anda menggunakan kunci privat untuk pasangan kunci baru dan hubungkan ke instans. Untuk informasi selengkapnya, lihat Hubungkan ke instans Windows Anda menggunakan RDP.

> **Important**
>
> Instans tersebut mendapatkan alamat IP publik baru setelah Anda menghentikan dan memulainya. Pastikan untuk terhubung ke instans menggunakan nama DNS publiknya saat ini. Untuk informasi selengkapnya, lihat Perubahan status instans Amazon EC2.

4. (Opsional) Jika Anda tidak menggunakan instans sementara lagi, Anda dapat mengakhirinya. Pilih instans sementara, dan pilih Status instans, Akhiri instans.

# Memecahkan masalah Sysprep dengan instans Amazon EC2 Windows

Jika Anda mengalami masalah atau menerima pesan kesalahan selama persiapan gambar, tinjau log berikut ini. Lokasi log berbeda-beda tergantung apakah Anda menjalankan EC2Config, EC2Lauch v1, atau EC2Launch v2 dengan Sysprep.

- `%WINDIR%\Panther\Unattendgc` (EC2Config, EC2Launch v1, dan EC2Launch v2)
- `%WINDIR%\System32\Sysprep\Panther` (EC2Config, EC2Launch v1, dan EC2Launch v2)
- `C:\Program Files\Amazon\Ec2ConfigService\Logs\Ec2ConfigLog.txt` (hanya EC2Config)
- `C:\ProgramData\Amazon\Ec2Config\Logs` (hanya EC2Config)
- `C:\ProgramData\Amazon\EC2-Windows\Launch\Log\EC2Launch.log` (hanya EC2Launch v1)

- `%ProgramData%\Amazon\EC2Launch\log\agent.log` (hanya EC2Launch v2)

Jika Anda menerima pesan kesalahan selama persiapan gambar dengan Sysprep, OS mungkin tidak dapat dijangkau. Untuk meninjau file log, Anda harus menghentikan instans, melampirkan volume root-nya ke instans sehat lainnya sebagai volume sekunder, lalu meninjau log yang disebutkan sebelumnya di volume sekunder. Untuk informasi selengkapnya tentang tujuan file log berdasarkan nama, lihat File Setup-Related Log Windows di dokumentasi Microsoft.

Jika Anda menemukan kesalahan di file log Unattendgc, gunakan Alat Pencarian Kesalahan Microsoft untuk mendapatkan detail selengkapnya tentang kesalahan tersebut. Masalah berikut yang dilaporkan di file log Unattendgc biasanya disebabkan oleh satu atau beberapa profil pengguna yang rusak pada instans:

```
Error [Shell Unattend] _FindLatestProfile failed (0x80070003) [gle=0x00000003]
Error [Shell Unattend] CopyProfile failed (0x80070003) [gle=0x00000003]
```

Ada dua opsi untuk menyelesaikan masalah ini:

Opsi 1

Gunakan Regedit pada instans untuk mencari kunci berikut. Verifikasi bahwa tidak ada kunci registri profil untuk pengguna yang dihapus.

`[HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\ProfileList\`

Opsi 2

1. Edit file yang relevan, sebagai berikut:
   - Windows Server 2012 R2 dan versi sebelumnya — Edit file jawaban EC2Config (`C:\Program Files\Amazon\Ec2ConfigService\sysprep2008.xml`).
   - Windows Server 2016 dan 2019 - Edit file jawaban unattend.xml (`C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep\Unattend.xml`).
   - Windows Server 2022 - Edit file jawaban unattend.xml (`C:\ProgramData\Amazon\EC2Launch\sysprep\unattend.xml`).
2. Ubah `<CopyProfile>true</CopyProfile>` ke `<CopyProfile>false</CopyProfile>`.
3. Jalankan lagi Syspre. Perhatikan bahwa perubahan konfigurasi ini akan menghapus profil pengguna administrator bawaan setelah Sysprep selesai.

# Memecahkan masalah instans Amazon EC2 Linux yang rusak menggunakan EC2Rescue

EC2Rescue untuk Linux adalah alat sumber terbuka yang mudah digunakan yang dapat dijalankan pada instans Amazon EC2 Linux untuk mendiagnosis, memecahkan masalah, dan memperbaiki masalah umum menggunakan pustaka lebih dari 100 modul. Modul adalah file YAML yang berisi skrip BASH atau Python dan metadata yang diperlukan.

Beberapa kasus penggunaan umum untuk EC2Rescue untuk instans Linux meliputi:

- Mengumpulkan log syslog dan manajer paket
- Mengumpulkan data pemanfaatan sumber daya
- Mendiagnosis dan memperbaiki parameter kernel bermasalah yang diketahui dan masalah OpenSSH umum

> **Note**
>
> Runbook `AWSSupport-TroubleshootSSH` AWS Systems Manager Otomasi menginstal EC2Rescue untuk Linux dan kemudian menggunakan alat untuk memeriksa atau mencoba memperbaiki masalah umum yang mencegah koneksi SSH ke instance Linux. Untuk informasi selengkapnya, lihat AWSSupport-TroubleshootSSH.

Jika Anda menggunakan instance Windows, lihatthe section called “EC2Rescue untuk instans Windows”.

Topik

- Instal EC2Rescue pada instans Amazon EC2 Linux
- Menjalankan perintah EC2Rescue pada instans Amazon EC2 Linux
- Kembangkan modul EC2Rescue untuk instans Amazon EC2 Linux

## Instal EC2Rescue pada instans Amazon EC2 Linux

Alat EC2Rescue untuk Linux dapat diinstal pada instans Amazon EC2 Linux yang memenuhi prasyarat berikut.

Prasyarat

- Sistem operasi yang didukung:
  - Amazon Linux 2
  - Amazon Linux 2016.09+
  - SUSE Linux Enterprise Server 12+
  - RHEL 7+
  - Ubuntu 16.04+
- Persyaratan perangkat lunak:
  - Python 2.7.9+ atau 3.2+

## Instal EC2Rescue

`AWSSupport-TroubleshootSSH`Runbook menginstal EC2Rescue untuk Linux dan kemudian menggunakan alat untuk memeriksa atau mencoba memperbaiki masalah umum yang mencegah koneksi jarak jauh ke mesin Linux melalui SSH. Untuk informasi lebih lanjut, dan untuk menjalankan otomatisasi ini, lihat Dukungan-TroubleshootSSH.

Jika sistem Anda memiliki versi Python yang diperlukan, Anda dapat menginstal build standar. Jika tidak, Anda dapat menginstal paket build, yang menyertakan salinan minimal Python.

Untuk menginstal build standar

1. Dari instans Linux yang berfungsi, unduh alat EC2Rescue untuk Linux:

   ```
   curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.tgz
   ```

2. (Opsional) Verifikasi tanda tangan file instalasi EC2Rescue untuk Linux. Untuk informasi selengkapnya, lihat (Opsional) Verifikasi tanda tangan EC2Resuce untuk Linux.
3. Unduh file sha256 hash:

   ```
   curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.tgz.sha256
   ```

4. Verifikasi integritas tarball:

   ```
   sha256sum -c ec2rl.tgz.sha256
   ```

5. Buka kemasan tarball:

```
tar -xzvf ec2rl.tgz
```

6. Verifikasi instalasi dengan mencantumkan file bantuan:

```
cd ec2rl-<version_number>
./ec2rl help
```

Untuk menginstal paket build

Untuk tautan ke unduhan dan daftar batasan, lihat EC2Rescue untuk Linux di github.

## (Opsional) Verifikasi tanda tangan EC2Resuce untuk Linux

Berikut ini adalah proses yang disarankan untuk memverifikasi validitas paket EC2Rescue untuk Linux untuk sistem Linux-based operasi.

Saat Anda mengunduh aplikasi dari internet, kami menyarankan Anda untuk mengautentikasi identitas penerbit perangkat lunak dan memeriksa apakah aplikasi tersebut belum diubah atau rusak setelah diterbitkan. Hal ini akan melindungi Anda agar tidak menginstal versi aplikasi yang berisi virus atau kode berbahaya lainnya.

Jika setelah menjalankan langkah-langkah dalam topik ini, Anda menganggap bahwa perangkat lunak untuk EC2Rescue untuk Linux telah diubah atau rusak, jangan menjalankan file instalasi. Alih-alih, hubungi Amazon Web Services.

EC2Rescue untuk file Linux untuk sistem Linux-based operasi ditandatangani menggunakan GnuPG, implementasi sumber terbuka dari standar Pretty Good Privacy (OpenPGP) untuk tanda tangan digital yang aman. GnuPG (juga dikenal sebagai GPG) menyediakan otentikasi dan pemeriksaan integritas melalui tanda tangan digital. AWS menerbitkan kunci publik dan tanda tangan yang dapat Anda gunakan untuk memverifikasi paket EC2Rescue untuk Linux yang diunduh. Untuk informasi selengkapnya tentang PGP dan GnuPG (GPG), lihat. https://www.gnupg.org/

Langkah pertamanya adalah membangun kepercayaan dengan penerbit perangkat lunak. Unduh kunci publik dari penerbit perangkat lunak, periksa apakah pemilik kunci publik adalah benar-benar pemiliknya, lalu tambahkan kunci publik ke dalam keyring Anda. Keyring adalah kumpulan kunci publik yang diketahui. Setelah menetapkan autentikasi kunci publik, Anda dapat menggunakannya untuk memverifikasi tanda tangan aplikasi.

Tugas

- Autentikasi dan impor kunci publik
- Memverifikasi tanda tangan paket

Autentikasi dan impor kunci publik

Langkah berikutnya dalam proses ini adalah mengautentikasi kunci publik EC2Rescue untuk Linux dan menambahkannya sebagai kunci tepercaya di dalam keyring GPG Anda.

Untuk mengautentikasi dan mengimpor kunci publik EC2Rescue untuk Linux

1. Pada prompt perintah, gunakan perintah berikut untuk mendapatkan salinan kunci publik build GPG kami:

   ```
   curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.key
   ```

2. Pada prompt perintah di direktori tempat Anda menyimpan `ec2rl.key`, gunakan perintah berikut untuk mengimpor kunci publik EC2Rescue untuk Linux ke dalam keyring Anda:

   ```
   gpg2 --import ec2rl.key
   ```

   Perintah tersebut akan mengembalikan hasil yang serupa dengan berikut ini:

   ```
   gpg: /home/ec2-user/.gnupg/trustdb.gpg: trustdb created
   gpg: key 2FAE2A1C: public key "ec2autodiag@amazon.com <EC2 Rescue for Linux>"
    imported
   gpg: Total number processed: 1
   gpg:               imported: 1  (RSA: 1)
   ```

   > **Tip**
   >
   > Jika Anda melihat kesalahan yang menyatakan bahwa perintah tidak dapat ditemukan, instal utilitas GnuPG dengan `apt-get install gnupg2` (Debian-based Linux) atau `yum install gnupg2` (Red Hat- berbasis Linux).

Memverifikasi tanda tangan paket

Setelah Anda menginstal alat GPG, mengautentikasi dan mengimpor kunci publik EC2Rescue untuk Linux, dan memverifikasi bahwa kunci publik EC2Rescue untuk Linux dapat dipercaya, Anda siap untuk memverifikasi tanda tangan skrip instalasi EC2Rescue untuk Linux.

Untuk memverifikasi tanda tangan skrip instalasi EC2Rescue untuk Linux

1. Pada prompt perintah, jalankan perintah berikut untuk mengunduh file tanda tangan untuk skrip instalasi:

   ```
   curl -O https://s3.amazonaws.com/ec2rescuelinux/ec2rl.tgz.sig
   ```

2. Verifikasi tanda tangan dengan menjalankan perintah berikut pada prompt perintah di direktori tempat Anda menyimpan `ec2rl.tgz.sig` dan file instalasi EC2Rescue untuk Linux. Kedua file tersebut harus ada.

   ```
   gpg2 --verify ./ec2rl.tgz.sig
   ```

   Output-nya akan terlihat seperti berikut ini:

   ```
   gpg: Signature made Thu 12 Jul 2018 01:57:51 AM UTC using RSA key ID 6991ED45
   gpg: Good signature from "ec2autodiag@amazon.com <EC2 Rescue for Linux>"
   gpg: WARNING: This key is not certified with a trusted signature!
   gpg:          There is no indication that the signature belongs to the owner.
   Primary key fingerprint: E528 BCC9 0DBF 5AFA 0F6C  C36A F780 4843 2FAE 2A1C
        Subkey fingerprint: 966B 0D27 85E9 AEEC 1146  7A9D 8851 1153 6991 ED45
   ```

   Jika output berisi frasa `Good signature from "ec2autodiag@amazon.com <EC2 Rescue for Linux>"`, ini berarti bahwa tanda tangan telah berhasil diverifikasi, dan Anda dapat menjalankan skrip instalasi EC2Rescue untuk Linux.

   Jika output berisi frasa `BAD signature`, periksa apakah Anda melakukan prosedur dengan benar. Jika Anda terus mendapatkan respons ini, hubungi Amazon Web Services dan jangan menjalankan file instalasi yang Anda unduh sebelumnya.

Berikut ini adalah informasi peringatan yang mungkin Anda lihat:

- WARNING: This key is not certified with a trusted signature! There is no indication that the signature belongs to the owner. Ini mengacu pada tingkat kepercayaan pribadi Anda dalam keyakinan bahwa Anda memiliki kunci publik yang autentik untuk EC2Rescue untuk Linux. Idealnya, Anda harus mendatangi kantor Amazon Web Services dan menerima kunci secara langsung. Namun, kemungkinan besar Anda akan mengunduhnya dari situs web. Dalam hal ini, situs web yang dimaksud adalah situs web Amazon Web Services .
- gpg2: no ultimately trusted keys found. Ini berarti bahwa kunci tersebut tidak "benar-benar tepercaya" oleh Anda (atau orang lain yang Anda percayai).

Untuk informasi selengkapnya, lihat https://www.gnupg.org/.

## Menjalankan perintah EC2Rescue pada instans Amazon EC2 Linux

EC2Rescue adalah alat baris perintah. Setelah Anda menginstal EC2Rescue pada instance Linux Anda, Anda bisa mendapatkan bantuan umum tentang cara menggunakan alat dengan menjalankannya. `./ec2rl help` Anda dapat melihat modul yang tersedia dengan menjalankan`./ec2rl list`, dan Anda bisa mendapatkan bantuan pada modul tertentu dengan menjalankan`./ec2rl help` *`module_name`*.

Berikut ini adalah tugas umum yang dapat Anda lakukan untuk mulai menggunakan alat ini.

Tugas

- Jalankan modul EC2Rescue
- Unggah hasil modul EC2Rescue
- Membuat cadangan instans Amazon EC2 Linux

### Jalankan modul EC2Rescue

Untuk menjalankan semua modul EC2Rescue

Gunakan ./ec2rl run perintah tanpa menentukan parameter tambahan apa pun. Beberapa modul memerlukan akses root. Jika Anda bukan pengguna root, gunakan sudo saat Anda menjalankan perintah.

```
./ec2rl run
```

Untuk menjalankan modul EC2Rescue tertentu

Gunakan ./ec2rl run perintah dan untuk`--only-modules`, tentukan nama modul yang akan dijalankan. Beberapa modul memerlukan argum en untuk menggunakannya.

```
./ec2rl run --only-modules=module_name --arguments
```

Misalnya, untuk menjalankan dig modul untuk menanyakan `amazon.com` domain, gunakan perintah berikut.

```
./ec2rl run --only-modules=dig --domain=amazon.com
```

Untuk melihat hasil modul EC2Rescue

Jalankan modul lalu lihat file log masuk`cat /var/tmp/ec2rl/`*`logfile_location`*. Misalnya, file log untuk dig modul dapat ditemukan di lokasi berikut:

```
cat /var/tmp/ec2rl/timestamp/mod_out/run/dig.log
```

## Unggah hasil modul EC2Rescue

Jika Dukungan telah meminta hasil untuk modul EC2Rescue, Anda dapat mengunggah file log menggunakan alat EC2Rescue. Anda dapat mengunggah hasil baik ke lokasi yang disediakan oleh Dukungan atau ke bucket Amazon S3 yang Anda miliki.

Untuk mengunggah hasil ke lokasi yang disediakan oleh Dukungan

Gunakan perintah ./ec2rl upload. Untuk`--upload-directory`, tentukan lokasi file log. Untuk`--support-url`, tentukan URL yang disediakan oleh Dukungan.

```
./ec2rl upload --upload-directory=/var/tmp/ec2rl/logfile_location --support-
url="url_provided_by_aws_support"
```

Untuk mengunggah hasil ke bucket Amazon S3

Gunakan perintah ./ec2rl upload. Untuk`--upload-directory`, tentukan lokasi file log. Untuk`--presigned-url`, tentukan URL yang telah ditandatangani sebelumnya untuk bucket S3. Untuk informasi selengkapnya tentang membuat URL yang telah ditandatangani sebelumnya untuk Amazon S3, lihat Meng unggah Objek Menggunakan Pre-Signed URL.

```
./ec2rl upload --upload-directory=/var/tmp/ec2rl/logfile_location --presigned-
url="presigned_s3_url"
```

### Membuat cadangan instans Amazon EC2 Linux

Anda dapat menggunakan EC2Rescue untuk membuat cadangan instance Linux Anda dengan membuat AMI atau dengan membuat snapshot dari volume terlampir.

Untuk membuat AMI

Gunakan `./ec2rl run` perintah dan untuk `--backup`, tentukan`ami`.

```
./ec2rl run --backup=ami
```

Untuk membuat snapshot multi-volume dari semua volume terlampir

Gunakan `./ec2rl run` perintah dan untuk `--backup`, tentukan`allvolumes`.

```
./ec2rl run --backup=allvolumes
```

Untuk membuat snapshot dari volume terlampir tertentu

Gunakan `./ec2rl run` perintah dan untuk `--backup`, tentukan ID volume yang akan dicadangkan.

```
./ec2rl run --backup=vol-01234567890abcdef
```

## Kembangkan modul EC2Rescue untuk instans Amazon EC2 Linux

Modul ditulis dalam YAML, yaitu standar serialisasi data. File YAML pada sebuah modul terdiri atas satu dokumen yang mewakili modul dan atributnya.

### Menambahkan atribut modul

Tabel berikut menjelaskan atribut modul yang tersedia.

| Atribut | Deskripsi |
| --- | --- |
| nama | Nama modul. Nama harus berisi 18 karakter atau kurang. |
| version | Nomor versi modul. |

| Atribut | Deskripsi |
| --- | --- |
| title | Judul singkat dan deskriptif untuk modul. Panjang judul harus berisi 50 karakter atau kurang. |
| helptext | Deskripsi modul yang lebih panjang. Setiap baris harus berisi 75 karakter atau kurang. Jika modul menggunakan argumen, yang bersifat wajib atau opsional, sertakan argumen tersebut dalam nilai teks bantuan.<br><br>Misalnya:<br><br>`helptext: !!str \|`<br>`  Collect output from ps for system analysis`<br>`  Consumes --times= for number of times to repeat`<br>`  Consumes --period= for time period between repetition` |
| placement | Tahap ketika modul harus dijalankan. Nilai yang didukung:<br><br>• prediagnostic<br>• run<br>• postdiagnostic |

| Atribut | Deskripsi |
| --- | --- |
| language | Bahasa yang digunakan untuk menuliskan kode modul. Nilai yang didukung:<br>• bash<br>• python<br><br>**Note**<br>Kode Python harus kompatibel dengan Python 2.7.9+ maupun Python 3.2+. |
| remediation | Menunjukkan apakah modul mendukung remediasi. Nilai yang didukung adalah `True` atau `False`.<br><br>Secara default, modul menjadi `False` jika hal ini tidak ada, sehingga membuatnya menjadi atribut opsional untuk modul yang tidak mendukung remediasi. |
| content | Keseluruhan kode skrip. |
| constraint | Nama objek yang berisi nilai kendala. |
| domain | Penjelasan tentang cara modul dikelompokkan atau diklasifikasikan. Rangkaian modul yang disertakan menggunakan domain berikut:<br>• application<br>• net<br>• os<br>• performance |

| Atribut | Deskripsi |
| --- | --- |
| class | Penjelasan tentang tipe tugas yang dilakukan oleh modul. Rangkaian modul yang disertakan menggunakan domain berikut:<br>• collect (mengambil output dari program)<br>• mendiagnosis (pass/fail berdasarkan serangkaian kriteria)<br>• gather (menyalin file dan menulis ke file tertentu) |
| distro | Daftar distribusi Linux yang didukung oleh modul ini. Rangkaian modul yang disertakan menggunakan distribusi berikut:<br>• alami (Amazon Linux)<br>• rhel<br>• ubuntu<br>• suse |
| diperlukan | Argumen wajib yang digunakan oleh modul dari opsi CLI. |
| optional | Argumen opsional yang dapat digunakan oleh modul. |
| software | Perangkat lunak yang dapat dieksekusi yang digunakan di dalam modul. Atribut ini ditujukan untuk menentukan perangkat lunak yang tidak terinstal secara default. Logika EC2Rescue untuk Linux memastikan bahwa program ini ada dan dapat dieksekusi sebelum menjalank an modul. |

| Atribut | Deskripsi |
| --- | --- |
| package | Paket perangkat lunak sumber untuk file yang dapat dieksekusi. Atribut ini ditujukan untuk memberikan detail tambahan pada paket dengan perangkat lunak, termasuk URL untuk mengunduh atau mendapatkan informasi lebih lanjut. |
| sudo | Mengindikasikan apakah akses root diperlukan untuk menjalankan modul.<br><br>Anda tidak perlu menerapkan pengecekan sudo dalam skrip modul. Jika nilainya benar, logika EC2Rescue untuk Linux hanya akan menjalankan modul ketika pengguna yang mengeksekusinya memiliki akses root. |
| perfimpact | Mengindikasikan apakah modul dapat menimbulkan dampak performa yang signifikan terhadap lingkungan tempat modul dijalankan. Jika nilainya benar dan argumen `--perfimpact=true` tidak ada, modul akan dilewati. |
| parallelexclusive | Menentukan program yang membutuhkan eksklusivitas bersama. Misalnya, semua modul yang menentukan proses "bpf" dijalankan secara bersambung. |

## Menambahkan variabel lingkungan

Tabel berikut menjelaskan variabel lingkungan yang tersedia.

| Variabel Lingkungan | Deskripsi |
| --- | --- |
| EC2RL_CALLPATH | Path ke ec2rl.py. Path ini dapat digunakan untuk menemukan direktori lib dan menggunakan modul Python yang bervendor. |
| EC2RL_WORKDIR | Direktori tmp utama untuk alat diagnostik.<br><br>Nilai default: /var/tmp/ec2rl . |
| EC2RL_RUNDIR | Direktori tempat semua output disimpan.<br><br>Nilai default: /var/tmp/ec2rl/<date&timestamp> . |
| EC2RL_GATHEREDDIR | Direktori root untuk menempatkan data modul yang dikumpulkan.<br><br>Nilai default:/var/tmp/ec2rl/<date&timestamp>/mod_out/gathered/ . |
| EC2RL_NET_DRIVER | Driver yang digunakan untuk antarmuka jaringan non-virtual pertama yang diurutkan menurut abjad pada instans.<br><br>Contoh:<br><br>• xen_netfront<br>• ixgbevf<br>• ena |
| EC2RL_SUDO | Betul jika EC2Rescue untuk Linux dijalankan sebagai root; jika tidak, salah. |
| EC2RL_VIRT_TYPE | Tipe virtualisasi seperti yang disediakan oleh metadata instans.<br><br>Contoh: |

| Variabel Lingkungan | Deskripsi |
| --- | --- |
| | • default-hvm<br>• default-paravirtual |
| `EC2RL_INTERFACES` | Daftar antarmuka yang disebutkan pada sistem. Nilai adalah string yang berisi nama, sepertieth0,eth1, dan seterusnya. Ini dihasilkan oleh `functions.bash` dan hanya tersedia untuk modul yang telah bersumber. |

## Menggunakan sintaks YAML

Hal-hal berikut ini harus diperhatikan saat menyusun file YAML modul Anda:

- Tiga tanda hubung (`---`) menunjukkan awal yang jelas dari suatu dokumen.
- Tanda `!ec2rlcore.module.Module` memberi tahu pengurai YAML konstruktor mana yang akan dipanggil saat membuat objek dari aliran data. Anda dapat menemukan konstruktor di dalam file `module.py`.
- Tanda `!!str` memberi tahu pengurai YAML untuk tidak mencoba menentukan tipe data, dan sebagai gantinya menginterpretasikan konten sebagai literal string.
- Simbol garis lurus (`|`) memberi tahu pengurai YAML bahwa nilainya adalah skalar gaya literal. Dalam hal ini, pengurai menyertakan semua spasi. Ini penting untuk modul karena indentasi dan karakter baris baru disimpan.
- Indentasi standar YAML adalah dua spasi, yang dapat dilihat pada contoh berikut. Pastikan Anda mempertahankan indentasi standar (misalnya, empat spasi untuk Python) untuk skrip Anda dan kemudian menunjukkan seluruh konten dua spasi di dalam file modul.

## Contoh modul

Contoh satu (`mod.d/ps.yaml`):

```
--- !ec2rlcore.module.Module
# Module document. Translates directly into an almost-complete Module object
name: !!str ps
path: !!str
version: !!str 1.0
```

```
title: !!str Collect output from ps for system analysis
helptext: !!str |
  Collect output from ps for system analysis
  Requires --times= for number of times to repeat
  Requires --period= for time period between repetition
placement: !!str run
package:
  - !!str
language: !!str bash
content: !!str |
  #!/bin/bash
  error_trap()
  {
      printf "%0.s=" {1..80}
      echo -e "\nERROR: "$BASH_COMMAND" exited with an error on line ${BASH_LINENO[0]}"
      exit 0
  }
  trap error_trap ERR

  # read-in shared function
  source functions.bash
  echo "I will collect ps output from this $EC2RL_DISTRO box for $times times every
 $period seconds."
  for i in $(seq 1 $times); do
      ps auxww
      sleep $period
  done
constraint:
  requires_ec2: !!str False
  domain: !!str performance
  class: !!str collect
  distro: !!str alami ubuntu rhel suse
  required: !!str period times
  optional: !!str
  software: !!str
  sudo: !!str False
  perfimpact: !!str False
  parallelexclusive: !!str
```

# Memecahkan masalah instans Windows Amazon EC2 yang rusak menggunakan EC2Rescue

EC2Rescue untuk Windows Server adalah alat yang mudah digunakan, yang Anda jalankan pada instans Windows Server Amazon EC2 untuk mendiagnosis dan memecahkan masalah yang mungkin terjadi. Ini berguna untuk mengumpulkan file log dan memecahkan masalah, juga mencari kemungkinan area yang menjadi permasalahan secara proaktif. Ini juga dapat memeriksa volume root Amazon EBS dari instans lain dan mengumpulkan log yang relevan untuk memecahkan masalah instans Windows Server menggunakan volume tersebut. Berikut ini adalah beberapa masalah umum yang dapat diatasi EC2Rescue:

- Masalah konektivitas instans karena firewall, Remote Desktop Protocol (RDP), atau konfigurasi antarmuka jaringan
- Masalah boot sistem operasi karena stop error, boot loop, atau registri yang rusak
- Masalah yang mungkin memerlukan analisis log lanjutan dan pemecahan masalah

EC2Rescue untuk Windows Server memiliki dua modul yang berbeda:

- Modul pengumpul data yang mengumpulkan data dari semua sumber yang berbeda
- Modul penganalisis yang mengurai data yang dikumpulkan terhadap serangkaian aturan yang telah ditentukan untuk mengidentifikasi masalah dan memberikan saran

Alat EC2Rescue untuk Windows Server hanya berjalan pada instans Amazon EC2 yang menjalankan Windows Server 2012 dan yang lebih baru. Saat dimulai, alat akan memeriksa apakah ia beroperasi di instans Amazon EC2 atau bukan.

> **Note**
>
> Runbook `AWSSupport-ExecuteEC2Rescue` AWS Systems Manager Otomasi menggunakan alat EC2Rescue untuk memecahkan masalah dan, jika memungkinkan, memperbaiki masalah konektivitas umum dengan instance EC2 yang ditentukan. Untuk informasi selengkapnya, dan untuk menjalankan otomatisasi ini, lihat > AWSSupport-ExecuteEC2Rescue.

Jika Anda menggunakan instance Linux, lihatthe section called “EC2Rescue untuk instans Linux”.

Topik


- Memecahkan masalah instans Windows yang rusak dengan EC2Rescue GUI
- Memecahkan masalah instans Windows yang rusak dengan EC2Rescue CLI
- Memecahkan masalah instans Windows yang rusak dengan EC2Rescue dan Systems Manager


# Memecahkan masalah instans Windows yang rusak dengan EC2Rescue GUI

EC2Rescue untuk Windows Server dapat melakukan analisis berikut pada instans  offline:

| Opsi | Deskripsi |
| --- | --- |
| Diagnosis dan Penyelamatan | EC2Rescue untuk Windows Server dapat mendeteksi dan mengatasi masalah dengan pengaturan layanan berikut ini:<br>• Waktu Sistem<br>  • RealTimeisUniversal - Mendeteksi apakah kunci registri `RealTimeisUniversal` diaktifkan. Jika dinonaktifkan, waktu sistem Windows menyimpang saat zona waktu disetel ke nilai selain UTC.<br>• Windows Firewall<br>  • Jaringan domain - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan.<br>  • Jaringan pribadi - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan.<br>  • Jaringan tamu atau publik - Mendeteksi apakah profil Windows Firewall ini diaktifka n atau dinonaktifkan. |

| Opsi | Deskripsi |
| --- | --- |
| | • Desktop Jarak Jauh<br>  • Mulai Layanan - Mendeteksi apakah layanan Desktop Jarak Jauh diaktifkan.<br>  • Koneksi Desktop Jarak Jauh - Mendeteksi apakah fitur ini diaktifkan.<br>  • Port TCP - Mendeteksi port mana yang mendengarkan layanan Desktop Jarak Jauh.<br><br>• EC2Config (Windows Server 2012 R2 dan versi sebelumnya)<br>  • Instalasi - Mendeteksi versi EC2Config apa yang diinstal.<br>  • Mulai Layanan - Mendeteksi apakah layanan EC2Config diaktifkan atau dinonaktifkan.<br>  • Ec2SetPassword - Membuat kata sandi administrator baru.<br>  • Ec2HandleUserData- Memungkinkan Anda menjalankan skrip data pengguna pada boot instance berikutnya.<br><br>• EC2Launch (Windows Server 2016 dan versi terbarunya)<br>  • Instalasi - Mendeteksi versi EC2Launch apa yang diinstal.<br>  • Ec2SetPassword - Membuat kata sandi administrator baru.<br><br>• Antarmuka Jaringan |

| Opsi | Deskripsi |
| --- | --- |
| | • Startup Layanan DHCP - Mendeteksi apakah layanan DHCP diaktifkan atau dinonaktifkan.<br>• Detail ethernet - Menampilkan informasi tentang versi driver jaringan, jika terdeteksi.<br>• DHCP di Ethernet - Mendeteksi apakah DHCP diaktifkan atau dinonaktifkan.<br>• Status tanda tangan disk<br>• Tanda tangan pada disk dan Tanda tangan pada Boot Configuration Database (BCD) - Mendeteksi apakah tanda tangan disk dan tanda tangan BCD sama atau tidak. Jika nilainya berbeda, EC2Rescue mencoba menimpa tanda tangan disk dengan tanda tangan pada BCD. |
| Pulihkan | Lakukan salah satu tindakan berikut:<br>• Konfigurasi Baik yang Terakhir Diketahui - Berupaya melakukan boot instans ke status terakhir yang diketahui dapat melakukan boot.<br>• Pulihkan registri dari cadangan - Memulihkan registri dari `\Windows\System32\config\RegBack` . |
| Menangkap Log | Memungkinkan Anda untuk menangkap log pada instans untuk analisis. |

EC2Rescue untuk Windows Server dapat mengumpulkan data berikut dari instans aktif dan offline:

| Item | Deskripsi |
| --- | --- |
| Log Peristiwa | Mengumpulkan log aplikasi, sistem, dan peristiwa EC2Config. |
| Registri | Mengumpulkan hive `SYSTEM` dan `SOFTWARE`. |
| Log Pembaruan Windows | Mengumpulkan file log yang dihasilkan oleh Pembaruan Windows.<br><br>**Note**<br>Di Windows Server 2016 dan versi yang lebih baru, log dikumpulkan dalam format Pelacakan Peristiwa untuk Windows (ETW). |
| Log Sysprep | Mengumpulkan file log yang dihasilkan oleh alat Windows System Preparation. |
| Log Pengaturan Driver | Mengumpulkan log Windows SetupAPI (`setupapi.dev.log` dan `setupapi.setup.log`). |
| Konfigurasi Boot | Mengumpulkan hive `HKEY_LOCAL_MACHINE\BCD00000000`. |
| Dump Memori | Mengumpulkan file dump memori yang ada pada instans. |
| File EC2Config | Mengumpulkan file log yang dihasilkan oleh layanan EC2Config. |
| File EC2Launch | Mengumpulkan file log yang dihasilkan oleh skrip EC2Launch. |
| File Agen SSM | Mengumpulkan file log yang dihasilkan oleh Agen SSM dan log Patch Manager. |

| Item | Deskripsi |
| --- | --- |
| File EC2 ElasticGPU | Mengumpulkan log peristiwa yang terkait dengan GPU elastic. |
| ECS | Kumpulkan log terkait ke Amazon ECS. |
| CloudEndure | Mengumpulkan file log yang terkait dengan CloudEndure Agen. |
| AWS Agen Replikasi untuk File Log MGN atau DRS | Mengumpulkan file log yang terkait dengan AWS Transform MGN atau AWS Elastic Disaster Recovery. |

EC2Rescue untuk Windows Server dapat mengumpulkan data tambahan berikut dari instans aktif:

| Item | Deskripsi |
| --- | --- |
| Informasi Sistem | Kumpulkan MSInfo32. |
| Hasil Kebijakan Grup | Mengumpulkan laporan Kebijakan Grup. |

## Analisis instans offline

Opsi Instans Offline berguna untuk mendebug permasalahan booting dengan instans Windows.

Untuk melakukan tindakan pada instans offline

1. Dari instans Windows Server yang berfungsi, unduh alat EC2Rescue untuk Windows Server dan ekstrak file.

   Anda dapat menjalankan PowerShell perintah berikut untuk mengunduh EC2Rescue tanpa mengubah Internet Explorer Enhanced Security Configuration (ESC) Anda:

   ```
   Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/EC2Rescue_latest.zip -
   OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
   ```

Perintah ini akan mengunduh file .zip EC2Rescue ke desktop pengguna yang saat ini masuk.

> **Note**
>
> Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:
>
> ```
> [Net.ServicePointManager]::SecurityProtocol =
>  [Net.SecurityProtocolType]::Tls12
> ```

2. Hentikan instans yang bermasalah, jika belum dihentikan.
3. Copot volume root EBS dari instans yang bermasalah dan lampirkan volume ke instans Windows yang berfungsi dan telah menginstal EC2Rescue untuk Windows Server.
4. Jalankan alat EC2Rescue untuk Windows Server pada instans yang berfungsi dan pilih Instans Offline.
5. Pilih disk volume yang baru dipasang dan pilih Berikutnya.
6. Konfirmasi pilihan disk dan pilih Ya.
7. Pilih opsi instans offline untuk dijalankan dan pilih Berikutnya.

Alat EC2Rescue untuk Windows Server memindai volume dan mengumpulkan informasi pemecahan masalah berdasarkan file log yang dipilih.

## Mengumpulkan data dari instans yang aktif

Anda dapat mengumpulkan log dan data lain dari instans yang aktif.

Untuk mengumpulkan data dari instans yang aktif

1. Hubungkan ke instans Windows Anda.
2. Unduh alat EC2Rescue untuk Windows Server ke instans Windows Anda dan ekstrak file.

   Anda dapat menjalankan PowerShell perintah berikut untuk mengunduh EC2Rescue tanpa mengubah Internet Explorer Enhanced Security Configuration (ESC) Anda:

```
Invoke-WebRequest https://s3.amazonaws.com/ec2rescue/windows/EC2Rescue_latest.zip -
OutFile $env:USERPROFILE\Desktop\EC2Rescue_latest.zip
```

Perintah ini akan mengunduh file .zip EC2Rescue ke desktop pengguna yang saat ini masuk.

> **Note**
>
> Jika Anda menerima kesalahan saat mengunduh file, dan Anda menggunakan Windows Server 2016 atau yang lebih lama, TLS 1.2 mungkin perlu diaktifkan untuk PowerShell terminal Anda. Anda dapat mengaktifkan TLS 1.2 untuk PowerShell sesi saat ini dengan perintah berikut dan kemudian coba lagi:
>
> ```
> [Net.ServicePointManager]::SecurityProtocol =
>  [Net.SecurityProtocolType]::Tls12
> ```

3. Buka aplikasi EC2Rescue untuk Windows Server dan terima perjanjian lisensi.
4. Pilih Berikutnya, Instans saat ini, Tangkap log.
5. Pilih item data yang akan dikumpulkan dan pilih Kumpulkan.... Baca peringatan dan pilih Ya untuk melanjutkan.
6. Pilih nama file dan lokasi untuk file ZIP, lalu pilih Simpan.
7. Setelah EC2Rescue untuk Windows Server selesai, pilih Buka Folder yang Berisi untuk melihat file ZIP.
8. Pilih Selesai.

## Memecahkan masalah instans Windows yang rusak dengan EC2Rescue CLI

EC2Rescue untuk antarmuka baris perintah (CLI) Windows Server memungkinkan Anda menjalankan plugin EC2Rescue untuk Windows Server (disebut sebagai “tindakan”) secara terprogram.

Alat EC2Rescue untuk Windows Server memiliki dua mode eksekusi:

- /online—Ini memungkinkan Anda untuk mengambil tindakan pada instans tempat EC2Rescue untuk Windows Server diinstal, seperti mengumpulkan file log.

- /offline:<device_id>—Ini memungkinkan Anda untuk mengambil tindakan pada volume root offline yang dilampirkan pada instans Windows Amazon EC2 terpisah, tempat Anda menginstal EC2Rescue untuk Windows Server.

Unduh alat EC2Rescue untuk Windows Server ke instans Windows Anda dan ekstrak file. Anda dapat melihat file bantuan menggunakan perintah berikut:

```
EC2RescueCmd.exe /help
```

EC2Rescue untuk Windows Server dapat melakukan tindakan berikut pada instans Windows Amazon EC2:

- Tindakan pengumpulan
- Tindakan penyelamatan
- Tindakan pemulihan

## Tindakan pengumpulan

> **Note**
>
> Anda dapat mengumpulkan semua log, seluruh grup log, atau satu log individu di dalam sebuah grup.

EC2Rescue untuk Windows Server dapat mengumpulkan data berikut dari instans aktif dan offline.

| Grup log | Log yang tersedia | Deskripsi |
| --- | --- | --- |
| `all` | | Kumpulkan semua log yang tersedia. |
| `eventlog` | • `'Application'`<br>• `'System'`<br>• `'EC2ConfigService'` | Mengumpulkan log aplikasi, sistem, dan peristiwa EC2Config. |

| Grup log | Log yang tersedia | Deskripsi |
| --- | --- | --- |
| memory-dump | • 'Memory Dump File'<br>• 'Mini Dump Files' | Mengumpulkan file pembuangan memori yang ada pada instans. |
| ec2config | • 'Log Files'<br>• 'Configuration Files' | Mengumpulkan file log yang dihasilkan oleh layanan EC2Config. |
| ec2launch | • 'Logs'<br>• 'Config' | Mengumpulkan file log yang dihasilkan oleh penulisan EC2Launch. |
| ssm-agent | • 'Log Files'<br>• 'Patch Baseline Logs'<br>• 'InstanceData' | Mengumpulkan file log yang dihasilkan oleh SSM Agent dan log Patch Manager. |
| sysprep | 'Log Files' | Mengumpulkan file log yang dihasilkan oleh alat Windows System Preparation. |
| driver-setup | • 'SetupAPI Log Files'<br>• 'DPInst Log File'<br>• 'AWS PV Setup Log File' | Mengumpulkan log Windows SetupAPI (setupapi.dev.log dan setupapi.setup.log ). |
| registry | • 'SYSTEM'<br>• 'SOFTWARE'<br>• 'BCD' | Mengumpulkan hive SYSTEM dan SOFTWARE. |
| egpu | • 'Event Log'<br>• 'System Files' | Mengumpulkan log peristiwa terkait GPU elastis. |
| boot-config | 'BCDEDIT Output' | Mengumpulkan hive HKEY_LOCAL_MACHINE\BCD00000000 . |

| Grup log | Log yang tersedia | Deskripsi |
| --- | --- | --- |
| `windows-update` | `'Log Files'` | Mengumpulkan file log yang dihasilkan oleh Windows Update.<br><br>**Note**<br>Di Windows Server 2016 dan lebih baru, log dikumpulkan dalam format Pelacakan Peristiwa untuk Windows (ETW). |
| `cloudendure` | • `'Migrate Script Logs'`<br>• `'Driver Logs'`<br>• `'CloudEndure File List'` | Mengumpulkan file log yang terkait dengan CloudEndure Agen. |

EC2Rescue untuk Windows Server dapat mengumpulkan data tambahan berikut dari instans aktif.

| Grup log | Log yang tersedia | Deskripsi |
| --- | --- | --- |
| `system-info` | `'MSInfo32 Output'` | Kumpulkan `MSInfo32`. |
| `gpresult` | `'GPResult Output'` | Mengumpulkan laporan Kebijakan Grup. |

Berikut ini adalah opsi yang tersedia:

- /output: <output FilePath > - Diperlukan lokasi jalur file tujuan untuk menyimpan file log yang dikumpulkan dalam format zip.

- /no-offline - Atribut opsional yang digunakan dalam mode offline. Tidak menetapkan volume secara offline setelah menyelesaikan tindakan.
- /no-fix-signature - Atribut opsional yang digunakan dalam mode offline. Tidak memperbaiki kemungkinan tabrakan tanda tangan disk setelah menyelesaikan tindakan.

Contoh

Berikut ini adalah contoh yang menggunakan EC2Rescue untuk CLI Windows Server.

Contoh mode online

Kumpulkan semua log yang tersedia:

```
EC2RescueCmd /accepteula /online /collect:all /output:<outputFilePath>
```

Hanya mengumpulkan grup log tertentu:

```
EC2RescueCmd /accepteula /online /collect:ec2config /output:<outputFilePath>
```

Kumpulkan log individu di dalam grup log:

```
EC2RescueCmd /accepteula /online /collect:'ec2config.Log Files,driver-setup.SetupAPI
 Log Files' /output:<outputFilePath>
```

Contoh mode offline

Kumpulkan semua log yang tersedia dari volume EBS. Volume ditentukan oleh nilai device_id.

```
EC2RescueCmd /accepteula /offline:xvdf /collect:all /output:<outputFilePath>
```

Hanya kumpulkan grup log tertentu:

```
EC2RescueCmd /accepteula /offline:xvdf /collect:ec2config /output:<outputFilePath>
```

## Tindakan penyelamatan

EC2Rescue untuk Windows Server dapat mendeteksi dan mengatasi masalah dengan pengaturan layanan berikut ini:

| Grup layanan | Tindakan yang tersedia | Deskripsi |
| --- | --- | --- |
| `all` | | |
| `system-time` | `'RealTimeIsUniversal'` | Waktu Sistem<br>• RealTimeisUniversal - Mendeteksi apakah kunci registri `RealTimeisUniversal` diaktifkan. Jika dinonaktifkan, waktu sistem Windows menyimpang saat zona waktu disetel ke nilai selain UTC. |
| `firewall` | • `'Domain networks'`<br>• `'Private networks'`<br>• `'Guest or public networks'` | Windows Firewall<br>• Jaringan domain - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan.<br>• Jaringan pribadi - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan.<br>• Jaringan tamu atau publik - Mendeteksi apakah profil Windows Firewall ini diaktifkan atau dinonaktifkan. |
| rdp | • `'Service Start'`<br>• `'Remote Desktop Connections'`<br>• `'TCP Port'` | Desktop Jarak Jauh<br>• Mulai Layanan - Mendeteksi apakah layanan Desktop Jarak Jauh diaktifkan. |

| Grup layanan | Tindakan yang tersedia | Deskripsi |
| --- | --- | --- |
| | | • Koneksi Desktop Jarak Jauh - Mendeteksi apakah fitur ini diaktifkan.<br>• Port TCP - Mendeteksi port mana yang mendengar kan layanan Desktop Jarak Jauh. |
| `ec2config` | • `'Service Start'`<br>• `'Ec2SetPassword'`<br>• `'Ec2HandleUserData'` | EC2Config<br>• Mulai Layanan - Mendeteksi apakah layanan EC2Config diaktifkan.<br>• Ec2SetPassword - Membuat kata sandi administrator baru.<br>• Ec2HandleUserData- Memungkinkan Anda menjalankan skrip data pengguna pada boot instance berikutnya. |
| `ec2launch` | `'Reset Administrator Password'` | Buat kata sandi administrator Windows baru. |
| `network` | `'DHCP Service Startup'` | Antarmuka Jaringan<br>• DHCP Service Startup - Mendeteksi apakah layanan DHCP diaktifkan. |

Berikut ini adalah opsi yang tersedia:

- /level:<level> - Atribut opsional untuk tingkat pemeriksaan yang harus dipicu oleh tindakan tersebut. Nilai yang diizinkan adalah: `information`, `warning`, `error`, `all`. Secara default, nilainya diatur ke `error`.
- /check-only - Atribut opsional yang menghasilkan laporan tetapi tidak melakukan modifikasi terhadap volume offline.

> **Note**
>
> Jika EC2Rescue untuk Windows Server mendeteksi kemungkinan tabrakan tanda tangan disk, itu akan mengoreksi tanda tangan setelah proses offline selesai secara default, bahkan ketika Anda menggunakan opsi tersebut. `/check-only` Anda harus menggunakan `/no-fix-signature` opsi untuk mencegah koreksi.

- /no-offline - Atribut opsional yang mencegah volume agar tidak diatur offline setelah menyelesaikan tindakan.
- /no-fix-signature - Atribut opsional yang tidak memperbaiki kemungkinan tabrakan tanda tangan disk setelah menyelesaikan tindakan.

Contoh penyelamatan

Berikut ini adalah contoh yang menggunakan EC2Rescue untuk Windows Server CLI. Volume ditentukan menggunakan nilai device_id.

Upayakan untuk memperbaiki semua masalah yang teridentifikasi pada volume:

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:all
```

Upayakan untuk memperbaiki semua masalah di dalam grup layanan pada volume:

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:firewall
```

Upayakan untuk memperbaiki item tertentu di dalam grup layanan pada volume:

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:rdp.'Service Start'
```

Tentukan banyak masalah yang akan dicoba diperbaiki pada volume:

```
EC2RescueCmd /accepteula /offline:xvdf /rescue:'system-
time.RealTimeIsUniversal,ec2config.Service Start'
```

## Tindakan pemulihan

EC2Rescue untuk Windows Server dapat mendeteksi dan mengatasi masalah dengan pengaturan layanan berikut ini:

| Grup Layanan | Tindakan yang Tersedia | Deskripsi |
|---|---|---|
| Memulihkan Konfigurasi Baik yang Terakhir Diketahui | `lkgc` | Konfigurasi Baik yang Terakhir Diketahui - Berupaya melakukan booting instans ke kondisi terakhir yang diketahui dapat melakukan boot. |
| Memulihkan registri Windows dari cadangan terbaru | `regback` | Pulihkan registri dari cadangan - Memulihkan registri dari `\Windows\System32\config\RegBack` . |

Berikut ini adalah opsi yang tersedia:

- /no-offline - Atribut opsional yang mencegah volume agar tidak diatur offline setelah menyelesaikan tindakan.
- /no-fix-signature - Atribut opsional yang tidak memperbaiki kemungkinan tabrakan tanda tangan disk setelah menyelesaikan tindakan.

### Contoh pemulihan

Berikut ini adalah contoh yang menggunakan EC2Rescue untuk Windows Server CLI. Volume ditentukan menggunakan nilai device_id.

Memulihkan konfigurasi baik yang terakhir diketahui pada volume:

```
EC2RescueCmd /accepteula /offline:xvdf /restore:lkgc
```

Pulihkan cadangan registri Windows terakhir pada volume:

```
EC2RescueCmd /accepteula /offline:xvdf /restore:regback
```

## Memecahkan masalah instans Windows yang rusak dengan EC2Rescue dan Systems Manager

Dukungan memberi Anda dokumen System Manager Run Command untuk berinteraksi dengan Manager-enabled instance Sistem Anda untuk menjalankan EC2Rescue untuk Windows Server. Dokumen Run Command ini disebut `AWSSupport-RunEC2RescueForWindowsTool`.

Dokumen Systems Manager Run Command ini melakukan tugas-tugas berikut:

- Unduh dan verifikasi EC2Rescue untuk Windows Server.
- Mengimpor PowerShell modul untuk memudahkan interaksi Anda dengan alat.
- Ber EC2RescueCmd jalan dengan perintah dan parameter yang disediakan.

Dokumen Systems Manager Run Command menerima tiga parameter:

- Perintah—EC2Rescue untuk tindakan Windows Server. Nilai-nilai yang diizinkan saat ini adalah:
  - ResetAccess—Mengatur ulang kata sandi Administrator lokal. Kata sandi Administrator lokal dari instans saat ini akan diatur ulang dan kata sandi yang dihasilkan secara acak akan disimpan dengan aman di Penyimpanan Parameter sebagai `/EC2Rescue/Password/<INSTANCE_ID>`. Jika Anda memilih tindakan ini dan tidak memberikan parameter, kata sandi akan dienkripsi secara otomatis dengan kunci KMS default. Secara opsional, Anda dapat menentukan ID Kunci KMS di Parameter untuk mengenkripsi kata sandi dengan kunci Anda sendiri.
  - CollectLogs—Menjalankan EC2Rescue untuk Windows Server dengan tindakan. `/collect:all` Jika Anda memilih tindakan ini, `Parameters` harus menyertakan nama bucket Amazon S3 untuk mengunggah log ke dalamnya.
  - FixAll—Menjalankan EC2Rescue untuk Windows Server dengan tindakan. `/rescue:all` Jika Anda memilih tindakan ini, `Parameters` harus menyertakan nama perangkat blok untuk penyelamatan.
- Parameter — PowerShell Parameter yang akan diteruskan untuk perintah yang ditentukan.

## Persyaratan

Untuk menjalankan  ResetAccess  tindakan, instans Amazon EC2 Anda harus memiliki kebijakan terlampir yang memberikan izin untuk menulis kata sandi terenkripsi ke Penyimpanan Parameter. Setelah melampirkan kebijakan, tunggu beberapa menit sebelum mencoba mengatur ulang kata sandi instance setelah Anda melampirkan kebijakan ini ke peran IAM terkait.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:PutParameter"
      ],
      "Resource": [
        "arn:aws:ssm:us-east-1:111122223333:parameter/EC2Rescue/
Passwords/<instanceid>"
      ]
    }
  ]
}
```

Jika Anda menggunakan kunci KMS khusus, bukan kunci KMS default, gunakan kebijakan ini sebagai gantinya.

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:PutParameter"
      ],
      "Resource": [
```

```
        "arn:aws:ssm:us-east-1:111122223333:parameter/EC2Rescue/
Passwords/<instanceid>"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:Encrypt"
      ],
      "Resource": [
        "arn:aws:kms:us-east-1:111122223333:key/<kmskeyid>"
      ]
    }
  ]
}
```

## Lihat JSON untuk dokumen

Prosedur berikut menjelaskan cara melihat JSON untuk dokumen ini.

Untuk menampilkan JSON pada dokumen Systems Manager Run Command

1. Buka AWS Systems Manager konsol di https://console.aws.amazon.com/systems-manager/.
2. Di panel navigasi, perluas Alat Manajemen Perubahan dan pilih Dok umen.
3. Di bilah pencarian, masukkan`AWSSupport-RunEC2RescueForWindowsTool`, lalu pilih `AWSSupport-RunEC2RescueForWindowsTool` dokumen.
4. Pilih tab Daftar Isi.

## Contoh

Berikut ini adalah beberapa contoh tentang cara menggunakan dokumen Systems Manager Run Command agar dapat menjalankan EC2Rescue untuk Windows Server, menggunakan AWS CLI. Untuk informasi selengkapnya tentang mengirim perintah menggunakan AWS CLI, lihat send-command.

Contoh

- Upayakan untuk memperbaiki semua masalah yang teridentifikasi pada volume root offline
- Kumpulkan log dari instans Windows Amazon EC2 saat ini

- Atur ulang kata sandi Administrator lokal

Upayakan untuk memperbaiki semua masalah yang teridentifikasi pada volume root offline

Upayakan untuk memperbaiki semua masalah yang teridentifikasi pada volume root offline yang dilampirkan pada instans Amazon EC2 Windows:

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --parameters "Command=FixAll, Parameters='xvdf'" --output
 text
```

Kumpulkan log dari instans Windows Amazon EC2 saat ini

Kumpulkan semua log dari instans Windows Amazon EC2 online saat ini dan unggah ke bucket Amazon S3:

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --parameters "Command=CollectLogs, Parameters='amzn-s3-
demo-bucket'" --output text
```

Atur ulang kata sandi Administrator lokal

Contoh berikut menunjukkan metode yang dapat Anda gunakan untuk mengatur ulang kata sandi Administrator lokal. Output ini menyediakan tautan ke Penyimpanan Parameter, di mana Anda dapat menemukan kata sandi yang aman dan dibuat secara acak, yang kemudian dapat Anda gunakan untuk RDP ke instans Windows Amazon EC2 sebagai Administrator lokal.

Setel ulang kata sandi Administrator lokal dari instance online menggunakan defaul AWS KMS key alias/aws t/ssm:

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --parameters "Command=ResetAccess" --output text
```

Atur ulang kata sandi Administrator lokal untuk instans online menggunakan kunci KMS:

```
aws ssm send-command --instance-ids "i-0cb2b964d3e14fd9f" --document-name "AWSSupport-
RunEC2RescueForWindowsTool" --parameters "Command=ResetAccess, Parameters=a133dc3c-
a2g4-4fc6-a873-6c0720104bf0" --output text
```

> **Note**
>
> Dalam contoh ini, kunci KMS adalah `a133dc3c-a2g4-4fc6-a873-6c0720104bf0`.

# EC2 Serial Console untuk instans

Dengan konsol serial EC2, Anda memiliki akses ke port serial instans Amazon EC2, yang dapat digunakan untuk memecahkan masalah boot, konfigurasi jaringan, dan masalah lainnya. Konsol serial tidak memerlukan instans Anda untuk memiliki kemampuan jaringan. Dengan konsol serial, Anda dapat memasukkan perintah ke sebuah instans seolah-olah keyboard dan monitor Anda terpasang secara langsung ke port serial instans. Sesi konsol serial berlangsung selama boot ulang dan penghentian instans. Selama boot ulang, Anda dapat melihat semua pesan boot dari awal.

Akses ke konsol serial tidak tersedia secara default. Organisasi Anda harus memberikan akses akun ke konsol serial dan mengonfigurasi kebijakan IAM untuk memberi pengguna akses ke konsol serial tersebut. Akses konsol serial dapat dikontrol pada tingkat terperinci dengan menggunakan ID instans, tanda sumber daya, dan lever IAM lainnya. Untuk informasi selengkapnya, lihat Konfigurasikan akses ke Konsol Serial EC2.

Konsol serial dapat diakses menggunakan konsol EC2 atau AWS CLI.

Konsol serial tersedia tanpa biaya tambahan.

Topik

- Prasyarat untuk Konsol Serial EC2
- Konfigurasikan akses ke Konsol Serial EC2
- Hubungkan ke Konsol Serial EC2
- Memutuskan koneksi dari Konsol Serial EC2
- Memecahkan masalah instans Amazon EC2 menggunakan Konsol Serial EC2

## Prasyarat untuk Konsol Serial EC2

Untuk menyambung ke Konsol Serial EC2 dan menggunakan alat pilihan Anda untuk pemecahan masalah, prasyarat berikut harus dipenuhi:

- Wilayah AWS

- Wavelength Zones
- AWS Outposts
- Zona Lokal
- Tipe instans
- Berikan akses
- Dukungan untuk klien berbasis peramban
- Status instans
- Amazon EC2 Systems Manager
- Konfigurasikan alat pemecahan masalah yang Anda pilih

### Wilayah AWS

Didukung di semua Wilayah AWS kecuali AWS European Sovereign Cloud (Germany).

### Wavelength Zones

Tidak didukung.

### AWS Outposts

Didukung di semua Pos AWS Terdepan.

### Zona Lokal

Didukung di semua Local Zones.

### Tipe instans

Tipe instans yang didukung:

- Linux
  - Semua instans virtual dibangun pada Nitro System.
  - Semua instans bare metal kecuali:
    - Tujuan umum:`a1.metal`,`mac1.metal`,`mac2.metal`,`mac2-m1ultra.metal`,`mac2-m2.metal`,`mac2-m2pro.metal`,`mac-m3ultra.metal`,`mac-m4.metal`,`mac-m4max.metal`, dan `mac-m4pro.metal`

    - Komputasi yang dipercepat: `g5g.metal`
    - Memori dioptimalkan: `u-6tb1.metalu-9tb1.metal`,`u-12tb1.metal`,,`u-18tb1.metal`, dan `u-24tb1.metal`
- Windows
    - Semua instans virtual dibangun pada Nitro System.
    - Semua instans bare metal kecuali:
        - Tujuan umum:`a1.metal`,`mac1.metal`,`mac2.metal`,`mac2-m1ultra.metal`,`mac2-m2.metal`,`mac2-m2pro.metal`,`mac-m3ultra.metal`,`mac-m4.metal`,`mac-m4max.metal`, dan `mac-m4pro.metal`
        - Komputasi yang dipercepat: `g5g.metal`
        - Memori dioptimalkan: `u-6tb1.metalu-9tb1.metal`,`u-12tb1.metal`,,`u-18tb1.metal`, dan `u-24tb1.metal`

## Berikan akses

Anda harus menyelesaikan tugas konfigurasi untuk memberikan akses ke Konsol Serial EC2. Untuk informasi selengkapnya, lihat Konfigurasikan akses ke Konsol Serial EC2.

## Dukungan untuk klien berbasis peramban

Untuk terhubung ke konsol serial menggunakan klien berbasis browser, browser Anda harus mendukung WebSocket. Jika browser Anda tidak mendukung WebSocket, sambungkan ke konsol serial menggunakan kunci Anda sendiri dan klien SSH.

Klien berbasis browser terhubung pada port 443 ke titik akhir proxy EC2 Instance Connect untuk Wilayah Anda. Format titik akhir tergantung pada partisi:

- Wilayah AWS Komersional dan Wilayah AWS GovCloud (AS) menggunakan format titik akhir:`prod.`*`region`*`.oneclickv2-proxy.ec2.aws.dev`.
- Wilayah China menggunakan format titik akhir:`prod.`*`region`*`.oneclickv2-proxy.ec2.a2z.org.cn`.

Contoh berikut menunjukkan titik akhir untuk setiap partisi:

- Untuk Wilayah Timur AS (Virginia Utara), titik akhir adalah`prod.us-east-1.oneclickv2-proxy.ec2.aws.dev`.

- Untuk Wilayah China (Beijing), titik akhirnya adalah`prod.cn-north-1.oneclickv2-proxy.ec2.a2z.org.cn`.

> **Note**
>
> Browser Anda harus dapat mencapai titik akhir ini. Jika firewall atau VPN memblokir koneksi, koneksi gagal.

### Status instans

Harus berupa `running`.

Anda tidak dapat terhubung ke konsol serial jika instans berada dalam status `pending`, `stopping`, `stopped`, `shutting-down`, atau `terminated`.

Untuk informasi selengkapnya tentang status instans, lihat Perubahan status instans Amazon EC2.

### Amazon EC2 Systems Manager

Jika instans menggunakan Amazon EC2 Systems Manager, Agen SSM versi 3.0.854.0 atau yang lebih baru harus diinstal pada instans tersebut. Untuk informasi tentang Agen SSM, lihat Bekerja dengan Agen SSM di Panduan Pengguna AWS Systems Manager .

### Konfigurasikan alat pemecahan masalah yang Anda pilih

Untuk memecahkan masalah instans menggunakan konsol serial, Anda dapat menggunakan GRUB atau SysRq pada instans Linux, dan Special Admin Console (SAC) pada instans Windows. Sebelum dapat menggunakan alat ini, Anda harus terlebih dahulu melakukan langkah-langkah konfigurasi pada setiap instans di tempat Anda akan menggunakannya.

Gunakan petunjuk untuk sistem operasi instans Anda untuk mengonfigurasi alat pemecahan masalah yang Anda pilih.

(Instans Linux) Konfigurasikan GRUB

Untuk mengonfigurasi GRUB, pilih salah satu dari prosedur berikut berdasarkan AMI yang digunakan untuk meluncurkan instans.

### Amazon Linux 2

Untuk mengonfigurasi GRUB pada instans Amazon Linux 2

1. Hubungkan ke instans Linux Anda, menggunakan SSH.
2. Tambahkan atau ubah pilihan berikut dalam `/etc/default/grub`:
   - Atur `GRUB_TIMEOUT=1`.
   - Tambahkan `GRUB_TERMINAL="console serial"`.
   - Tambahkan `GRUB_SERIAL_COMMAND="serial --speed=115200"`.

   Berikut adalah contoh `/etc/default/grub`. Anda mungkin perlu mengubah konfigurasi berdasarkan pengaturan sistem.

   ```
   GRUB_CMDLINE_LINUX_DEFAULT="console=tty0 console=ttyS0,115200n8 net.ifnames=0
    biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff rd.shell=0"
   GRUB_TIMEOUT=1
   GRUB_DISABLE_RECOVERY="true"
   GRUB_TERMINAL="console serial"
   GRUB_SERIAL_COMMAND="serial --speed=115200"
   ```
3. Terapkan konfigurasi yang diperbarui dengan menjalankan perintah berikut.

   ```
   [ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
   ```

### Ubuntu

Untuk mengonfigurasi GRUB pada instans Ubuntu

1. Terhubung ke instans Anda.
2. Tambahkan atau ubah pilihan berikut dalam `/etc/default/grub.d/50-cloudimg-settings.cfg`:
   - Atur `GRUB_TIMEOUT=1`.
   - Tambahkan `GRUB_TIMEOUT_STYLE=menu`.
   - Tambahkan `GRUB_TERMINAL="console serial"`.
   - Hapus `GRUB_HIDDEN_TIMEOUT`.

- Tambahkan `GRUB_SERIAL_COMMAND="serial --speed=115200"`.

Berikut adalah contoh `/etc/default/grub.d/50-cloudimg-settings.cfg`. Anda mungkin perlu mengubah konfigurasi berdasarkan pengaturan sistem.

```
# Cloud Image specific Grub settings for Generic Cloud Images
# CLOUD_IMG: This file was created/modified by the Cloud Image build process

# Set the recordfail timeout
GRUB_RECORDFAIL_TIMEOUT=0

# Do not wait on grub prompt
GRUB_TIMEOUT=1
GRUB_TIMEOUT_STYLE=menu

# Set the default commandline
GRUB_CMDLINE_LINUX_DEFAULT="console=tty1 console=ttyS0
 nvme_core.io_timeout=4294967295"

# Set the grub console type
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed 115200"
```

3. Terapkan konfigurasi yang diperbarui dengan menjalankan perintah berikut.

```
[ec2-user ~]$ sudo update-grub
```

RHEL

Untuk mengonfigurasi GRUB pada instans RHEL

1. Sambungkan ke instans Anda.
2. Tambahkan atau ubah pilihan berikut dalam `/etc/default/grub`:
   - Hapus `GRUB_TERMINAL_OUTPUT`.
   - Tambahkan `GRUB_TERMINAL="console serial"`.
   - Tambahkan `GRUB_SERIAL_COMMAND="serial --speed=115200"`.

Berikut adalah contoh `/etc/default/grub`. Anda mungkin perlu mengubah konfigurasi berdasarkan pengaturan sistem.

```
GRUB_TIMEOUT=1
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS0,115200n8 net.ifnames=0
 rd.blacklist=nouveau nvme_core.io_timeout=4294967295 crashkernel=auto"
GRUB_DISABLE_RECOVERY="true"
GRUB_ENABLE_BLSCFG=true
GRUB_TERMINAL="console serial"
GRUB_SERIAL_COMMAND="serial --speed=115200"
```

3. Terapkan konfigurasi yang diperbarui dengan menjalankan perintah berikut.

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg --update-bls-cmdline
```

Untuk RHEL 9.2 dan sebelumnya, gunakan perintah berikut.

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

### CentOS

Untuk instans yang diluncurkan menggunakan CentOS AMI, GRUB dikonfigurasikan untuk konsol serial secara default.

Berikut adalah contoh `/etc/default/grub`. Konfigurasi Anda mungkin berbeda berdasarkan pengaturan sistem.

```
GRUB_TIMEOUT=1
GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL="serial console"
GRUB_SERIAL_COMMAND="serial --speed=115200"
GRUB_CMDLINE_LINUX="console=tty0 crashkernel=auto console=ttyS0,115200"
GRUB_DISABLE_RECOVERY="true"
```

(Instans Linux) Konfigurasikan SysRq

Untuk mengkonfigurasi SysRq, Anda mengaktifkan SysRq perintah untuk siklus boot saat ini. Untuk membuat konfigurasi tetap, Anda juga dapat mengaktifkan SysRq perintah untuk boot berikutnya.

Untuk mengaktifkan semua SysRq perintah untuk siklus boot saat ini

1. Sambungkan ke instans Anda.
2. Jalankan perintah berikut.

   ```
   [ec2-user ~]$ sudo sysctl -w kernel.sysrq=1
   ```

   Pengaturan ini akan dihapus pada boot ulang berikutnya.

Untuk mengaktifkan semua SysRq perintah untuk boot berikutnya

1. Buat file `/etc/sysctl.d/99-sysrq.conf` dan buka file tersebut di editor favorit Anda.

   ```
   [ec2-user ~]$ sudo vi /etc/sysctl.d/99-sysrq.conf
   ```
2. Tambahkan baris berikut.

   ```
   kernel.sysrq=1
   ```
3. Boot ulang instans untuk menerapkan perubahan.

   ```
   [ec2-user ~]$ sudo reboot
   ```
4. Pada `login` prompt, masukkan nama pengguna berbasis kata sandi yang Anda atur sebelumnya, lalu tekan Enter.
5. Pada perintah `Password`, masukkan kata sandi, lalu tekan Enter.

(Instans Windows) Aktifkan SAC dan menu boot

> **Note**
>
> Jika Anda mengaktifkan SAC pada instans, layanan EC2 yang mengandalkan pengambilan kata sandi tidak akan bekerja dari konsol Amazon EC2. Agen peluncuran Windows di Amazon EC2 (EC2Config, EC2Launch v1, dan EC2Launch v2) mengandalkan konsol

serial untuk menjalankan berbagai tugas. Tugas-tugas ini tidak berhasil dijalankan saat Anda mengaktifkan SAC pada sebuah instans. Untuk informasi selengkapnya tentang agen peluncuran Windows di Amazon EC2, lihatthe section called “Konfigurasikan instance Windows”. Jika mengaktifkan SAC, Anda dapat menonaktifkannya nanti. Untuk informasi selengkapnya, lihat Menonaktifkan SAC dan menu boot.

Gunakan salah satu metode berikut untuk mengaktifkan SAC dan menu boot pada instans.

PowerShell

Untuk mengaktifkan SAC dan menu boot pada instans Windows

1. Hu bungkan ke instance Anda dan lakukan langkah-langkah berikut dari baris PowerShell perintah yang ditinggikan.
2. Aktifkan SAC.

```
bcdedit /ems '{current}' on
bcdedit /emssettings EMSPORT:1 EMSBAUDRATE:115200
```

3. Aktifkan menu boot.

```
bcdedit /set '{bootmgr}' displaybootmenu yes
bcdedit /set '{bootmgr}' timeout 15
bcdedit /set '{bootmgr}' bootems yes
```

4. Terapkan konfigurasi yang diperbarui dengan melakukan boot ulang instans.

```
shutdown -r -t 0
```

Command prompt

Untuk mengaktifkan SAC dan menu boot pada instans Windows

1. Hubungkan ke instans Anda dan lakukan langkah-langkah berikut dari prompt perintah.
2. Aktifkan SAC.

```
bcdedit /ems {current} on
bcdedit /emssettings EMSPORT:1 EMSBAUDRATE:115200
```

3. Aktifkan menu boot.

```
bcdedit /set {bootmgr} displaybootmenu yes
bcdedit /set {bootmgr} timeout 15
bcdedit /set {bootmgr} bootems yes
```

4. Terapkan konfigurasi yang diperbarui dengan melakukan boot ulang instans.

```
shutdown -r -t 0
```

## Konfigurasikan akses ke Konsol Serial EC2

Untuk mengonfigurasi akses ke konsol serial, Anda harus memberikan akses konsol serial pada tingkat akun, lalu mengonfigurasi kebijakan IAM untuk memberikan akses kepada pengguna IAM. Untuk instans Linux, Anda juga harus mengonfigurasi pengguna berbasis kata sandi pada setiap instance sehingga pengguna Anda dapat menggunakan konsol serial untuk pemecahan masalah.

EC2 Serial Console menggunakan koneksi port serial virtual untuk berinteraksi dengan instance. Koneksi ini independen dari instance VPC, sehingga Anda dapat bekerja dengan sistem operasi instance dan menjalankan alat pemecahan masalah meskipun ada kegagalan boot atau masalah konfigurasi jaringan. Karena koneksi ini berada di luar jaringan VPC, EC2 Serial Console tidak menggunakan grup keamanan instans atau ACL jaringan subnet untuk mengotorisasi lalu lintas ke instans.

Sebelum Anda mulai

Pastikan prasyarat terpenuhi.

Daftar Isi


- Tingkat akses ke Konsol Serial EC2
- Kelola akses akun ke Konsol Serial EC2
- Konfigurasikan kebijakan IAM untuk akses Konsol Serial EC2
- Mengatur kata sandi pengguna OS pada instance Linux

## Tingkat akses ke Konsol Serial EC2

Secara default, tidak ada akses ke konsol serial pada tingkat akun. Anda perlu secara eksplisit memberikan akses ke konsol serial pada tingkat akun. Untuk informasi selengkapnya, lihat Kelola akses akun ke Konsol Serial EC2.

Anda dapat menggunakan kebijakan kontrol layanan (SCP) untuk mengizinkan akses ke konsol serial dalam organisasi. Anda selanjutnya dapat memiliki kontrol akses terperinci pada tingkat pengguna menggunakan kebijakan IAM untuk mengontrol akses. Dengan menggunakan kombinasi kebijakan SCP dan IAM, Anda memiliki beragam tingkat kontrol akses ke konsol serial.

Tingkat organisasi

Anda dapat menggunakan kebijakan kontrol layanan (SCP) untuk mengizinkan akses ke konsol serial di organisasi Anda. Untuk informasi selengkapnya tentang SCP, lihat Kebijakan kontrol layanan di Panduan Pengguna AWS Organizations .

Tingkat instans

Anda dapat mengonfigurasi kebijakan akses konsol serial dengan menggunakan IAM PrincipalTag dan ResourceTag konstruksi dan dengan menentukan instance berdasarkan ID-nya. Untuk informasi selengkapnya, lihat Konfigurasikan kebijakan IAM untuk akses Konsol Serial EC2.

Tingkat pengguna

Anda dapat mengonfigurasi akses pada tingkat pengguna dengan mengonfigurasi kebijakan IAM untuk mengizinkan atau menolak pengguna tertentu izin guna mendorong kunci publik SSH ke layanan konsol serial instans tertentu. Untuk informasi selengkapnya, lihat Konfigurasikan kebijakan IAM untuk akses Konsol Serial EC2.

Level OS (hanya instans Linux)

Anda dapat mengatur kata sandi pengguna pada tingkat OS tamu. Tingkat ini menyediakan akses ke konsol serial untuk beberapa kasus penggunaan. Namun, untuk memantau log, Anda tidak memerlukan pengguna berbasis kata sandi. Untuk informasi selengkapnya, lihat Mengatur kata sandi pengguna OS pada instance Linux.

## Kelola akses akun ke Konsol Serial EC2

Secara default, tidak ada akses ke konsol serial pada tingkat akun. Anda perlu secara eksplisit memberikan akses ke konsol serial pada tingkat akun.

Pengaturan ini dikonfigurasi di tingkat akun, baik secara langsung di akun atau dengan menggunakan kebijakan deklaratif. Itu harus dikonfigurasi di setiap Wilayah AWS tempat Anda ingin memberikan akses ke konsol serial. Menggunakan kebijakan deklaratif memungkinkan Anda menerapkan pengaturan di beberapa Wilayah secara bersamaan, serta di beberapa akun secara bersamaan. Saat kebijakan deklaratif digunakan, Anda tidak dapat mengubah setelan secara langsung di dalam akun. Topik ini menjelaskan cara mengonfigurasi pengaturan secara langsung di dalam akun. Untuk informasi tentang penggunaan kebijakan deklaratif, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Daftar Isi


- Memberikan izin kepada pengguna untuk mengelola akses akun
- Melihat status akses akun ke konsol serial
- Memberikan akses akun ke konsol serial
- Menolak akses akun ke konsol serial


## Memberikan izin kepada pengguna untuk mengelola akses akun

Untuk mengizinkan pengguna mengelola akses akun ke konsol serial EC2, Anda perlu memberi mereka izin IAM yang diperlukan.

Kebijakan berikut memberikan izin untuk melihat status akun serta untuk mengizinkan dan mencegah akses akun ke konsol serial EC2.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ec2:GetSerialConsoleAccessStatus",
                "ec2:EnableSerialConsoleAccess",
                "ec2:DisableSerialConsoleAccess"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

Untuk informasi selengkapnya, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Melihat status akses akun ke konsol serial

Console

Untuk melihat akses akun ke konsol serial

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Peng aturan, pilih Konsol Serial EC2.
4. Pada tab Konsol Serial EC2, nilai akses Konsol Serial EC2 adalah Di izinkan atau Dicegah.

AWS CLI

Untuk melihat akses akun ke konsol serial

Gunakan perintah get-serial-console-access-status.

```
aws ec2 get-serial-console-access-status
```

Berikut ini adalah output contoh. Nilai `true` menunjukkan bahwa akun diizinkan akses ke konsol serial.

```
{
    "SerialConsoleAccessEnabled": true,
    "ManagedBy": "account"
}
```

Bid `ManagedBy` ang menunjukkan entitas yang mengkonfigurasi pengaturan. Nilai yang mungkin adalah `account` (dikonfigurasi secara langsung) atau`declarative-policy`. Untuk informasi selengkapnya, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

PowerShell

Untuk melihat akses akun ke konsol serial

Gunakan Get-EC2SerialConsoleAccessStatus cmdlet.

```
Get-EC2SerialConsoleAccessStatus -Select *
```

Berikut ini adalah output contoh. Nilai `True` menunjukkan bahwa akun diizinkan akses ke konsol serial.

```
ManagedBy SerialConsoleAccessEnabled
--------- -------------------------
account   True
```

Bid `ManagedBy` ang menunjukkan entitas yang mengkonfigurasi pengaturan. Nilai yang mungkin adalah `account` (dikonfigurasi secara langsung) atau`declarative-policy`. Untuk informasi selengkapnya, lihat Kebijakan deklaratif di Panduan AWS Organizations Pengguna.

Memberikan akses akun ke konsol serial

Console

Untuk memberikan akses akun ke konsol serial

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Peng aturan, pilih Konsol Serial EC2.
4. Pilih Kelola.
5. Untuk mengizinkan akses ke konsol serial EC2 dari semua instans di akun, pilih kotak centang Iz inkan.
6. Pilih Perbarui.

AWS CLI

Untuk memberikan akses akun ke konsol serial

Gunakan perintah enable- serial-console-access.

```
aws ec2 enable-serial-console-access
```

Berikut ini adalah output contoh.

```
{
    "SerialConsoleAccessEnabled": true
}
```

PowerShell

Untuk memberikan akses akun ke konsol serial

Gunakan Enable-EC2SerialConsoleAccess cmdlet.

```
Enable-EC2SerialConsoleAccess
```

Berikut ini adalah output contoh.

```
True
```

Menolak akses akun ke konsol serial

Console

Untuk menolak akses akun ke konsol serial

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Dasbor.
3. Pada kartu Atribut Akun, di bawah Peng aturan, pilih Konsol Serial EC2.
4. Pilih Kelola.
5. Untuk mencegah akses ke konsol serial EC2 dari semua instans di akun, kosongkan kotak centang Iz inkan.
6. Pilih Perbarui.

AWS CLI

Untuk menolak akses akun ke konsol serial

Gunakan perintah disable- serial-console-access.

```
aws ec2 disable-serial-console-access
```

Berikut ini adalah output contoh.

```
{
    "SerialConsoleAccessEnabled": false
}
```

PowerShell

Untuk menolak akses akun ke konsol serial

Gunakan Disable-EC2SerialConsoleAccess cmdlet.

```
Disable-EC2SerialConsoleAccess
```

Berikut ini adalah output contoh.

```
False
```

## Konfigurasikan kebijakan IAM untuk akses Konsol Serial EC2

Secara default, pengguna Anda tidak memiliki akses ke konsol serial. Organisasi Anda harus mengonfigurasi kebijakan IAM untuk memberikan akses yang diperlukan kepada pengguna. Untuk informasi selengkapnya, lihat Membuat kebijakan IAM dalam Panduan Pengguna IAM.

Untuk akses konsol serial, buat dokumen kebijakan JSON yang mencakup tindakan `ec2-instance-connect:SendSerialConsoleSSHPublicKey`. Tindakan ini memberi pengguna izin untuk mendorong kunci publik ke layanan konsol serial, yang memulai sesi konsol serial. Sebaiknya batasi akses ke instans EC2 tertentu. Jika tidak, semua pengguna IAM dengan izin ini dapat terhubung ke konsol serial dari semua instans EC2.

Contoh kebijakan IAM

- Secara eksplisit mengizinkan akses ke konsol serial
- Secara eksplisit mengizinkan akses ke konsol serial
- Gunakan tanda sumber daya untuk mengontrol akses ke konsol serial

Secara eksplisit mengizinkan akses ke konsol serial

Secara default, tidak ada yang memiliki akses ke konsol serial. Untuk memberikan akses ke konsol serial, Anda perlu mengonfigurasi kebijakan untuk secara eksplisit mengizinkan akses. Sebaiknya konfigurasikan kebijakan yang membatasi akses ke instans tertentu.

Kebijakan berikut memungkinkan akses ke konsol serial instans tertentu, diidentifikasi berdasarkan ID instansnya.

Perhatikan bahwa tindakan `DescribeInstances`, `DescribeInstanceTypes`, dan `GetSerialConsoleAccessStatus` tidak mendukung izin tingkat sumber daya, dan oleh karena itu semua sumber daya, yang ditunjukkan oleh `*` (tanda bintang), harus ditentukan untuk tindakan ini.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowDescribeInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceTypes",
                "ec2:GetSerialConsoleAccessStatus"
            ],
             "Resource": "*"
        },
        {
            "Sid": "AllowinstanceBasedSerialConsoleAccess",
            "Effect": "Allow",
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
            ],
            "Resource": "arn:aws:ec2:us-
east-1:111122223333:instance/i-0598c7d356eba48d7"
        }
    ]
}
```

Secara eksplisit mengizinkan akses ke konsol serial

Kebijakan IAM berikut memungkinkan akses ke konsol serial semua instans, dilambangkan dengan * (tanda bintang), dan secara eksplisit menolak akses ke konsol serial instans tertentu, diidentifikasi berdasarkan ID-nya.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSerialConsoleAccess",
            "Effect": "Allow",
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey",
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceTypes",
                "ec2:GetSerialConsoleAccessStatus"
            ],
            "Resource": "*"
        },
        {
            "Sid": "DenySerialConsoleAccess",
            "Effect": "Deny",
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
            ],
            "Resource": "arn:aws:ec2:us-
east-1:111122223333:instance/i-0598c7d356eba48d7"
        }
    ]
}
```

Gunakan tanda sumber daya untuk mengontrol akses ke konsol serial

Anda dapat menggunakan tanda sumber daya untuk mengontrol akses ke konsol serial dari sebuah instans.

Attribute-based kontrol akses adalah strategi otorisasi yang mendefinisikan izin berdasarkan tag yang dapat dilampirkan ke pengguna dan AWS sumber daya. Misalnya, kebijakan berikut ini mengizinkan

pengguna untuk memulai koneksi konsol serial untuk sebuah instans hanya jika tanda sumber daya instans dan tanda pengguna utama memiliki nilai `SerialConsole` yang sama untuk kunci tanda tersebut.

Untuk informasi selengkapnya tentang menggunakan tag untuk mengontrol akses ke AWS sumber daya Anda, lihat Meng ontrol akses ke AWS sumber daya di Panduan Pengguna IAM.

Perhatikan bahwa tindakan `DescribeInstances`, `DescribeInstanceTypes`, dan `GetSerialConsoleAccessStatus` tidak mendukung izin tingkat sumber daya, dan oleh karena itu semua sumber daya, yang ditunjukkan oleh * (tanda bintang), harus ditentukan untuk tindakan ini.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "AllowDescribeInstances",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeInstances",
                "ec2:DescribeInstanceTypes",
                "ec2:GetSerialConsoleAccessStatus"
            ],
            "Resource": "*"
        },
        {
            "Sid": "AllowTagBasedSerialConsoleAccess",
            "Effect": "Allow",
            "Action": [
                "ec2-instance-connect:SendSerialConsoleSSHPublicKey"
            ],
            "Resource": "arn:aws:ec2:us-east-1:111122223333:instance/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/SerialConsole":
 "${aws:PrincipalTag/SerialConsole}"
                }
            }
        }
    ]
}
```

## Mengatur kata sandi pengguna OS pada instance Linux

Anda dapat terhubung ke konsol serial tanpa kata sandi. Namun, untuk menggunakan konsol serial untuk memecahkan masalah instance Linux, instance harus memiliki pengguna OS berbasis kata sandi.

Anda dapat mengatur kata sandi untuk pengguna OS apa pun, termasuk pengguna root. Perhatikan bahwa pengguna root dapat memodifikasi semua file, sementara setiap pengguna OS mungkin memiliki izin terbatas.

Anda harus mengatur kata sandi pengguna pada setiap instans untuk konsol serial yang akan Anda gunakan. Ini adalah persyaratan satu kali untuk setiap instans.

> **Note**
>
> Secara default, AMI yang disediakan oleh tidak AWS dikonfigurasi dengan pengguna berbasis kata sandi. Jika Anda meluncurkan instans menggunakan AMI yang sudah memiliki kata sandi pengguna root yang dikonfigurasikan, Anda dapat melewati instruksi ini.

Untuk mengatur kata sandi pengguna OS pada instance Linux

1. Terhubung ke instans Anda. Anda dapat menggunakan metode apa pun untuk terhubung ke instans, kecuali metode koneksi Konsol Serial EC2.
2. Untuk mengatur kata sandi pengguna, gunakan perintah passwd. Pada contoh berikut, pengguna adalah `root`.

   ```
   [ec2-user ~]$ sudo passwd root
   ```

   Berikut ini adalah output contoh.

   ```
   Changing password for user root.
   New password:
   ```
3. Pada perintah `New password`, masukkan kata sandi baru.
4. Pada perintah, masukkan kembali kata sandinya.

# Hubungkan ke Konsol Serial EC2

Anda dapat menyambung ke konsol serial instans EC2 Anda dengan menggunakan konsol Amazon EC2 atau melalui SSH. Setelah terhubung ke konsol serial, Anda dapat menggunakannya untuk memecahkan masalah boot, konfigurasi jaringan, dan masalah lainnya. Untuk informasi selengkapnya tentang pemecahan masalah, lihat Memecahkan masalah instans Amazon EC2 menggunakan Konsol Serial EC2.

Pertimbangan-pertimbangan

- Hanya 1 koneksi konsol serial aktif yang didukung per instans.
- Koneksi konsol serial biasanya berlangsung selama 1 jam kecuali Anda memutuskan sambungannya. Namun, selama pemeliharaan sistem, Amazon EC2 akan memutuskan koneksi sesi konsol serial.

  Durasi koneksi tidak ditentukan oleh durasi kredenSIAL IAM Anda. Jika kredensi IAM Anda kedaluwarsa, koneksi akan terus bertahan hingga durasi maksimum koneksi konsol serial tercapai. Saat menggunakan pengalaman konsol Konsol Serial EC2, jika kredenSIAL IAM Anda kedaluwarsa, hentikan koneksi dengan menutup halaman browser.
- Diperlukan waktu 30 detik untuk meruntuhkan sesi setelah Anda memutuskan koneksi dari konsol serial untuk memungkinkan sesi baru.
- Port konsol serial yang didukung: `ttyS0` (Instans Linux) dan `COM1` (Instans Windows)
- Saat terhubung ke konsol serial, Anda mungkin melihat sedikit penurunan throughput instans.

Topik

- Hubungkan menggunakan klien berbasis peramban
- Hubungkan menggunakan kunci Anda sendiri dan klien SSH
- Titik akhir dan sidik jari Konsol Serial EC2

## Hubungkan menggunakan klien berbasis peramban

Anda dapat terhubung ke konsol serial instans EC2 menggunakan klien berbasis peramban. Anda melakukannya dengan memilih instans di konsol Amazon EC2 dan memilih untuk terhubung ke konsol serial. Klien berbasis peramban menangani izin dan menyediakan koneksi yang berhasil.

Konsol serial EC2 bekerja dari sebagian besar peramban dan mendukung input keyboard serta mouse.

Sebelum menghubungkan, pastikan Anda telah menyelesaikan prasyarat.

Untuk terhubung ke port serial instans Anda menggunakan klien berbasis browser (konsol Amazon EC2)

1. Buka konsol Amazon EC2 di https://console.aws.amazon.com/ec2/.
2. Di panel navigasi, pilih Instans.
3. Pilih instans dan pilih Tindakan, Pantau dan pecahkan masalah, Konsol Serial EC2, Hubungkan.

   Atau, pilih instans dan pilih Hubungkan, Konsol Serial EC2, Hubungkan.

   Jendela terminal dalam peramban akan terbuka.
4. Tekan Enter. Jika perintah login ditampilkan, Anda terhubung ke konsol serial.

   Jika layar tetap berwarna hitam, Anda dapat menggunakan informasi berikut untuk membantu menyelesaikan masalah saat menghubungkan ke konsol serial:

   - Pastikan bahwa Anda telah mengonfigurasi akses ke konsol serial. Untuk informasi selengkapnya, lihat Konfigurasikan akses ke Konsol Serial EC2.
   - (Hanya instans Linux) Gunakan SysRq untuk menyambung ke konsol serial. SysRq tidak mengharuskan Anda terhubung menggunakan klien berbasis browser. Untuk informasi selengkapnya, lihat (Instans Linux) Gunakan SysRq untuk memecahkan masalah instans Anda.
   - (Hanya instans Linux) Mulai ulang Getty. Jika Anda memiliki akses SSH ke instans, hubungkan ke instans menggunakan SSH, dan mulai ulang getty menggunakan perintah berikut.

     ```
     [ec2-user ~]$ sudo systemctl restart serial-getty@ttyS0
     ```
   - Boot ulang instans Anda. Anda dapat me-reboot instans Anda dengan menggunakan SysRq (instans Linux), konsol EC2, atau. AWS CLI Untuk informasi selengkapnya, lihat (Instans Linux) Gunakan SysRq untuk memecahkan masalah instans Anda (Instans Linux) atauMulai ulang instans Amazon EC2 Anda.
5. (Hanya instans Linux) Pada `login` prompt, masukkan nama pengguna berbasis kata sandi yang Anda atur sebelumnya, lalu tekan Enter.

6. (Hanya instans Linux) Pada `Password` prompt, masukkan kata sandi, lalu tekan Enter.

## Hubungkan menggunakan kunci Anda sendiri dan klien SSH

Anda dapat menggunakan kunci SSH Anda sendiri dan terhubung ke instans dari klien SSH pilihan Anda saat menggunakan API konsol serial. Hal ini memungkinkan Anda untuk mendapatkan manfaat dari kemampuan konsol serial untuk mendorong kunci publik ke instans.

Setelah menekan kunci SSH ke instance, koneksi SSH tidak tunduk pada kebijakan IAM yang Anda konfigurasi untuk memberikan akses Konsol Serial EC2 kepada pengguna.

Sebelum Anda mulai

Pastikan prasyarat terpenuhi.

Untuk terhubung ke konsol serial instans menggunakan SSH

1. Dorong kunci publik SSH Anda ke instans untuk memulai sesi konsol serial

   Gunakan perintah send-serial-console-ssh-public-key untuk mendorong kunci publik SSH Anda ke instans. Tindakan tersebut akan memulai sesi konsol serial.

   Jika sesi konsol serial telah dimulai untuk instans ini, perintah menjadi gagal karena Anda hanya dapat memiliki satu sesi terbuka pada satu waktu. Diperlukan waktu 30 detik untuk meruntuhkan sesi setelah Anda memutuskan koneksi dari konsol serial untuk memungkinkan sesi baru.

   ```
   aws ec2-instance-connect send-serial-console-ssh-public-key \
       --instance-id i-001234a4bf70dec41EXAMPLE \
       --serial-port 0 \
       --ssh-public-key file://my_key.pub \
       --region us-east-1
   ```

2. Hubungkan ke konsol serial menggunakan kunci privat Anda

   Gunakan perintah ssh untuk terhubung ke konsol serial sebelum kunci publik dihapus dari layanan konsol serial. Anda memiliki waktu 60 detik sebelum kunci publik dihapus.

   Gunakan kunci privat yang sesuai dengan kunci publik.

   Format nama pengguna adalah`instance-id.port0`, yang terdiri dari ID instance dan port 0. Dalam contoh berikut, nama pengguna adalah`i-001234a4bf70dec41EXAMPLE.port0`.

Titik akhir layanan konsol serial berbeda untuk setiap Wilayah. Lihat tabel Titik akhir dan sidik jari Konsol Serial EC2 untuk setiap titik akhir Wilayah. Pada contoh berikut, layanan konsol serial berada di Wilayah us-east-1.

```
ssh -i my_key i-001234a4bf70dec41EXAMPLE.port0@serial-console.ec2-instance-connect.us-east-1.aws
```

Contoh berikut digunakan `timeout 3600` untuk mengatur sesi SSH Anda untuk berakhir setelah 1 jam. Proses yang dimulai selama sesi dapat terus berjalan di instans Anda setelah sesi berakhir.

```
timeout 3600 ssh -i my_key i-001234a4bf70dec41EXAMPLE.port0@serial-console.ec2-instance-connect.us-east-1.aws
```

3. (Opsional) Verifikasi sidik jari

   Saat terhubung ke konsol serial untuk pertama kalinya, Anda akan diminta untuk memverifikasi sidik jari. Anda dapat membandingkan sidik jari konsol serial dengan sidik jari yang ditampilkan untuk verifikasi. Jika sidik jari ini tidak cocok, seseorang mungkin mencoba melakukan serangan “man-in-the-middle”. Jika sidik jari cocok, Anda dapat terhubung ke konsol serial dengan yakin.

   Sidik jari berikut ditujukan untuk layanan konsol serial di Wilayah us-east-1. Untuk sidik jari setiap Wilayah, lihat Titik akhir dan sidik jari Konsol Serial EC2.

   ```
   SHA256:dXwn5ma/xadVMeBZGEru5l2gx+yI5LDiJaLUcz0FMmw
   ```

   Sidik jari hanya muncul saat pertama kali Anda terhubung ke konsol serial.

4. Tekan Enter. Jika perintah kembali, Anda terhubung ke konsol serial.

   Jika layar tetap berwarna hitam, Anda dapat menggunakan informasi berikut untuk membantu menyelesaikan masalah saat menghubungkan ke konsol serial:

   - Pastikan bahwa Anda telah mengonfigurasi akses ke konsol serial. Untuk informasi selengkapnya, lihat Konfigurasikan akses ke Konsol Serial EC2.
   - (Hanya instans Linux) Gunakan SysRq untuk menyambung ke konsol serial. SysRq tidak mengharuskan Anda terhubung menggunakan SSH. Untuk informasi selengkapnya, lihat (Instans Linux) Gunakan SysRq untuk memecahkan masalah instans Anda.

- (Hanya instans Linux) Mulai ulang Getty. Jika Anda memiliki akses SSH ke instans, hubungkan ke instans menggunakan SSH, dan mulai ulang getty menggunakan perintah berikut.

```
[ec2-user ~]$ sudo systemctl restart serial-getty@ttyS0
```

- Boot ulang instans Anda. Anda dapat me-reboot instans Anda dengan menggunakan SysRq (hanya instans Linux), konsol EC2, atau. AWS CLI Untuk informasi selengkapnya, lihat (Instans Linux) Gunakan SysRq untuk memecahkan masalah instans Anda (hanya instans Linux) atauMulai ulang instans Amazon EC2 Anda.

5. (Hanya instans Linux) Pada `login` prompt, masukkan nama pengguna berbasis kata sandi yang Anda atur sebelumnya, lalu tekan Enter.
6. (Hanya instans Linux) Pada `Password` prompt, masukkan kata sandi, lalu tekan Enter.

## Titik akhir dan sidik jari Konsol Serial EC2

Berikut ini adalah titik akhir layanan dan sidik jari untuk Konsol Serial EC2. Untuk terhubung secara terprogram ke konsol serial instans, Anda menggunakan titik akhir Konsol Serial EC2. Titik akhir dan sidik jari Konsol Serial EC2 bersifat unik untuk setiap Wilayah AWS .

| Nama Wilayah | Wilayah | Titik akhir | Sidik jari |
|---|---|---|---|
| AS Timur (Ohio) | us-east-2 | serial-console.ec2-instance-connect.us-east-2.aws | SHA256:EhwPkTzRtTY7TRSzz26XbB0/HvV9jRM7mCZN0xw/d/0 |
| US East (Northern Virginia) | us-east-1 | serial-console.ec2-instance-connect.us-east-1.aws | SHA256:dXwn5ma/xadVMeBZGEru5l2gx+y I5LDiJaLUcz0FMmw |
| AS Barat (California Utara) | us-west-1 | serial-console.ec2-instance-connect.us-west-1.aws | SHA256:OHldlcMET8u7QLSX3jmR |

| Nama Wilayah | Wilayah | Titik akhir | Sidik jari |
|---|---|---|---|
| | | | TRAPFHVtq byoLZBMUCqiH3Y |
| AS Barat (Oregon) | us-west-2 | serial-console.ec2-instance-connect.us-west-2.aws | SHA256:EM CIe23TqKaBI6yGHain qZcMwqNkD hhAVHa1O2JxVUc |
| Africa (Cape Town) | af-south-1 | ec2-serial-console.af-south-1.api.aws | SHA256:RM WWZ2fVePe JUqzjO5jL2KIgXsczo Hlz21Ed00biiWI |
| Asia Pasifik (Hong Kong) | ap-east-1 | ec2-serial-console.ap-east-1.api.aws | SHA256:T0Q1lpiXxCh oZHplnAkjbP7tkm2xX ViC9bJFsjYnifk |
| Asia Pasifik (Taipei) | ap-east-2 | ec2-serial-console.ap-east-2.api.aws | SHA256:z1 rsF5DE8aQqsHwBr/ D40tR2GnyMq nScCUa1z1eFwDQ |
| Asia Pasifik (Hyderabad) | ap-south-2 | ec2-serial-console.ap-south-2.api.aws | SHA256:WJ gPBSwV4/shN +OPITValoewAuYj1 5DVW845JEhDKRs |
| Asia Pasifik (Jakarta) | ap-southeast-3 | ec2-serial-console.ap-southeast-3.api.aws | SHA256:5ZwgrCh+lfn s32XITqL/4O0zIfbx4 bZgsYFqy3o8mIk |
| Asia Pasifik (Malaysia) | ap-southeast-5 | ec2-serial-console.ap-Southeast-5.api.aws | SHA256:cQ XTHQMRcqR dIjmAGoAM BSExeoRob YyRwT67yTjnEiA |

| Nama Wilayah | Wilayah | Titik akhir | Sidik jari |
|---|---|---|---|
| Asia Pasifik (Selandia Baru) | ap-Southeast-6 | ec2-serial-console.ap-Southeast-6.api.aws | SHA256:wN ltdH0gVfM5uHJKjrwE lFuKoJgalcSKqjCS+L LkV 4 |
| Asia Pacific (Melbourne) | ap-southeast-4 | ec2-serial-console.ap-southeast-4.api.aws | SHA256:Av aq27hFgLv jn5gTSShZ 0oV7h90p0 GG46wfOeT6ZJvM |
| Asia Pasifik (Mumbai) | ap-south-1 | serial-console.ec2-instance-connect.ap-south-1.aws | SHA256:oB LXcYmklqH HEbliARxEgH8IsO51r ezTPiSM35BsU40 |
| Asia Pacific (Osaka) | ap-northeast-3 | ec2-serial-console.ap-northeast-3.api.aws | SHA256:Am0/ jiBKBnBuFnHr9aXs gEV3G8Tu/ vVHFXE/3UcyjsQ |
| Asia Pasifik (Seoul) | ap-northeast-2 | serial-console.ec2-instance-connect.ap-northeast-2.aws | SHA256:FoqWXNX +DZ++ +frczm2ds Rqri GuNTztg9P K49WYMqBX |
| Asia Pasifik (Singapura) | ap-southeast-1 | serial-console.ec2-instance-connect.ap-southeast-1.aws | SHA256:PL FNn7WnCQD Hx3qmwLu1Gy/ O8TUX7LQgZuaC6L 45CoY |
| Asia Pasifik (Sydney) | ap-southeast-2 | serial-console.ec2-instance-connect.ap-southeast-2.aws | SHA256:yF vMwUK9lEU QjQTRoXXzuN+cW9/ VSe9W984Cf5Tgzo4 |

| Nama Wilayah | Wilayah | Titik akhir | Sidik jari |
| --- | --- | --- | --- |
| Asia Pasifik (Thailand) | ap-Southeast-7 | ec2-serial-console.ap-Southeast-7.api.aws | SHA256:KC AZiRYrR1Q 2lqsg7vTw ixWmvc2wm jVT31XRgSdEfDY |
| Asia Pasifik (Tokyo) | ap-northeast-1 | serial-console.ec2-instance-connect.ap-northeast-1.aws | SHA256:RQ fsDCZTOfQ awewTRDV1t9Em/ HMrFQe+Crliot5um4k |
| Kanada (Pusat) | ca-central-1 | serial-console.ec2-instance-connect.ca-central-1.aws | SHA256:P2 O2jOZwmpM wkpO6YW73 8FIOTHdUT yEv2gczYMMO7s4 |
| Kanada Barat (Calgary) | ca-west-1 | ec2-serial-console.ca-barat-1.api.aws | SHA256:s3 rc8lI2xhbhr3iedjJN xGAFLPGOL jx7IxxXrGckk6Q |
| Tiongkok (Beijing) | cn-north-1 | ec2-serial-console .cn-north-1.api.am azonwebservices.co m.cn | SHA256:2g HVFy4H7uU3+ WaFUxD28v/ ggMeqjvSlgngpgLgG T +Y |
| Tiongkok (Ningxia) | cn-northwest-1 | ec2-serial-console .cn-northwest-1.ap i.amazonwebservice s.com.cn | SHA256:Td grNZkiQOd VfYEBUhO4 SzUA09VWI 5rYOZGTogpwmiM |

| Nama Wilayah | Wilayah | Titik akhir | Sidik jari |
|---|---|---|---|
| Eropa (Frankfurt) | eu-central-1 | serial-console.ec2-instance-connect.eu-central-1.aws | SHA256:aCMFS/yIcOdOlkXvOl8AmZ1Toe+b BnrJJ3Fy0k0De2c |
| Eropa (Irlandia) | eu-west-1 | serial-console.ec2-instance-connect.eu-west-1.aws | SHA256:h2AaGAWO4Hathhtm6ezs3Bj7udgUxi2qTrHjZAwCW6E |
| Eropa (London) | eu-west-2 | serial-console.ec2-instance-connect.eu-west-2.aws | SHA256:a69rd5CE/AEG4Amm53I6lkD1ZPvS/BCV3tTPW2RnJg8 |
| Europe (Milan) | eu-south-1 | ec2-serial-console.eu-south-1.api.aws | SHA256:IC0kOVJnpgFyBVrxn0A7n99ecLbXSX95cuuS7X7QK30 |
| Eropa (Paris) | eu-west-3 | serial-console.ec2-instance-connect.eu-west-3.aws | SHA256:q8ldnAf9pymeNe8BnFVngY3RPAr/kxswJUzfrlxeEWs |
| Eropa (Spanyol) | eu-south-2 | ec2-serial-console.eu-south-2.api.aws | SHA256:GoCW2DFRlu669QNxqFxEcsR6fZUz/4F4n7T45ZcwoEc |
| Eropa (Stockholm) | eu-north-1 | serial-console.ec2-instance-connect.eu-north-1.aws | SHA256:tkGFFUVUDvocDiGSS3Cu8Gdl6w2uI32EPNpKFKLwX84 |

| Nama Wilayah | Wilayah | Titik akhir | Sidik jari |
|---|---|---|---|
| Europe (Zurich) | eu-central-2 | ec2-serial-console.eu-central-2.api.aws | SHA256:8Ppx2mBMf6WdCw0NUlzKfwM4/IfRz4OaXFutQXWp6mk |
| Israel (Tel Aviv) | il-central-1 | ec2-serial-console.il-central-1.api.aws | SHA256:JR6q8v6kNNPi8+QSFQ4dj5dimNmZPTgwgsM1SNvtYyU |
| Meksiko (Tengah) | mx-sentral-1 | ec2-serial-konsol.mx-central-1.api.aws | SHA256:BCuVl13iQNk+CcVnt18Ef4p2ZHUrBBAOxlFetB32GS0 |
| Timur Tengah (Bahrain) | me-south-1 | ec2-serial-console.me-south-1.api.aws | SHA256:nPjLLKHu2QnLdUq2kVArsoK5xvPJOMRJKCBzCDqC3k8 |
| Timur Tengah (UAE) | eu-central-1 | ec2-serial-console.me-central-1.api.aws | SHA256:zpb5duKiBZ+l0dFwPeyykB4MPBYhI/XzXNeFSDKBvLE |
| Amerika Selatan (Sao Paulo) | sa-east-1 | serial-console.ec2-instance-connect.sa-east-1.aws | SHA256:rd2+/32+ OgnjewlyVIemENaQzCBotbih62OqAPDq1dI |
| AWS GovCloud (US-East) | us-gov-east-1 | serial-console.ec2-instance-connect.us-gov-east-1.amazonaws.com | SHA256:tIwe19GWsoyLClrtvu38YEEh+DHIkqnDcZnmtebvF28 |

| Nama Wilayah | Wilayah | Titik akhir | Sidik jari |
| --- | --- | --- | --- |
| AWS GovCloud (US-West) | us-gov-west-1 | serial-console.ec2-instance-connect.us-gov-west-1.amazonaws.com | SHA256:kfOFRWLaOZfB+utbd3b Rf8OlPf8nGO2YZLqXZiIw5DQ |

## Memutuskan koneksi dari Konsol Serial EC2

Jika tidak perlu lagi terhubung ke Konsol Serial EC2, Anda dapat memutuskan koneksi konsol tersebut. Saat Anda memutuskan koneksi dari konsol serial, sesi shell apa pun yang berjalan pada instans akan terus berjalan. Jika Anda ingin mengakhiri sesi shell, Anda harus mengakhirinya sebelum memutuskan koneksi dari konsol serial.

Pertimbangan-pertimbangan

- Koneksi konsol serial biasanya berlangsung selama 1 jam kecuali jika Anda memutuskan koneksi dari konsol tersebut. Namun, selama pemeliharaan sistem, Amazon EC2 akan memutuskan koneksi sesi konsol serial.
- Diperlukan waktu 30 detik untuk meruntuhkan sesi setelah Anda memutuskan koneksi dari konsol serial untuk memungkinkan sesi baru.

Cara untuk memutuskan koneksi dari konsol serial bergantung pada klien.

Browser-based klien

Untuk memutuskan koneksi dari konsol serial, tutup jendela terminal dalam peramban konsol serial.

Klien OpenSSH standar

Untuk memutuskan koneksi dari konsol serial, gunakan perintah berikut untuk menutup koneksi SSH. Perintah ini harus dijalankan segera setelah baris baru.

```
~.
```

Perintah yang digunakan untuk menutup koneksi SSH mungkin berbeda bergantung pada klien SSH yang Anda gunakan.

# Memecahkan masalah instans Amazon EC2 menggunakan Konsol Serial EC2

Dengan menggunakan Konsol Serial EC2, Anda dapat memecahkan masalah boot, konfigurasi jaringan, dan masalah lainnya dengan terhubung ke port serial instans.

Gunakan instruksi untuk sistem operasi instans Anda dan untuk alat yang telah Anda konfigurasikan pada instans Anda.

Alat

- (Instans Linux) Gunakan GRUB untuk memecahkan masalah instans Anda
- (Instans Linux) Gunakan SysRq untuk memecahkan masalah instans Anda
- (Instans Windows) Gunakan SAC untuk memecahkan masalah instans Anda

Prasyarat

Sebelum memulai, pastikan Anda telah menyelesaikan prasyarat, termasuk mengonfigurasi alat pemecahan masalah yang Anda pilih.

## (Instans Linux) Gunakan GRUB untuk memecahkan masalah instans Anda

GNU GRUB (singkatan dari GNU GRand Unified Bootloader, yang biasa disebut GRUB) adalah loader boot default untuk sebagian besar sistem operasi Linux. Dari menu GRUB, Anda dapat memilih kernel mana yang akan di-boot, atau memodifikasi entri menu untuk mengubah cara kernel akan di-boot. Hal ini dapat berguna ketika memecahkan masalah kegagalan instans.

Menu GRUB ditampilkan selama proses boot. Menu tidak dapat diakses melalui SSH normal, tetapi Anda dapat mengaksesnya menggunakan Konsol Serial EC2.

Anda dapat melakukan boot ke mode pengguna tunggal atau mode darurat. Mode pengguna tunggal akan melakukan boot ulang kernel pada runlevel yang lebih rendah. Misalnya, mode ini mungkin memasang sistem, file tetapi tidak mengaktifkan jaringan, sehingga memberi Anda kesempatan untuk melakukan pemeliharaan yang diperlukan guna memperbaiki instans. Mode darurat mirip dengan mode pengguna tunggal kecuali kernel berjalan pada runlevel terendah.

Untuk melakukan boot ke mode pengguna tunggal

1. Hubungkan ke konsol serial instans.

2. Boot ulang instans menggunakan perintah berikut.

```
[ec2-user ~]$ sudo reboot
```

3. Selama boot ulang berlangsung, saat menu GRUB muncul, tekan tombol apa pun untuk menghentikan proses boot.
4. Pada menu GRUB, gunakan tombol panah untuk memilih kernel guna melakukan boot, dan tekan e di keyboard Anda.
5. Gunakan tombol panah untuk meletakkan kursor Anda pada baris yang berisi kernel. Garis dimulai dengan `linux` atau `linux16` bergantung pada AMI yang digunakan untuk meluncurkan instans. Untuk Ubuntu, dua baris dimulai dengan `linux`, yang keduanya harus dimodifikasi pada langkah berikutnya.
6. Di akhir baris, tambahkan kata `single`.

   Berikut ini adalah contoh Amazon Linux 2.

```
linux /boot/vmlinuz-4.14.193-149.317.amzn2.aarch64 root=UUID=d33f9c9a-\
dadd-4499-938d-ebbf42c3e499 ro  console=tty0 console=ttyS0,115200n8 net.ifname\
s=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff rd.she\
ll=0 single
```

7. Tekan Ctrl+X untuk melakukan boot ke mode pengguna tunggal.
8. Pada `login` prompt, masukkan nama pengguna berbasis kata sandi yang Anda atur sebelumnya, lalu tekan Enter.
9. Pada perintah `Password`, masukkan kata sandi, lalu tekan Enter.

Untuk mem-boot ke mode darurat

Ikuti langkah-langkah yang sama seperti mode pengguna tunggal, tetapi pada langkah 6, tambahkan kata alih- `emergency` alih`single`.

## (Instans Linux) Gunakan SysRq untuk memecahkan masalah instans Anda

Kunci System Request (SysRq), yang kadang-kadang disebut sebagai “magic SysRq “, dapat digunakan untuk mengirim perintah langsung ke kernel, di luar shell, dan kernel akan merespons, terlepas dari apa yang dilakukan kernel. Misalnya, jika instance berhenti merespons, Anda dapat

menggunakan SysRq kunci untuk memberi tahu kernel agar crash atau reboot. Untuk informasi lebih lanjut, lihat K SysRq unci ajaib di Wikipedia.

Anda dapat menggunakan SysRq perintah di klien berbasis browser EC2 Serial Console atau di klien SSH. Perintah untuk mengirim permintaan jeda berbeda untuk setiap klien.

Untuk menggunakan SysRq, pilih salah satu prosedur berikut berdasarkan klien yang Anda gunakan.

Browser-based client

Untuk digunakan SysRq di klien berbasis browser konsol serial

1. Hubungkan ke konsol serial instans.
2. Untuk mengirim permintaan jeda, tekan `CTRL+0` (nol). Jika keyboard mendukungnya, Anda juga dapat mengirim permintaan jeda menggunakan tombol Pause atau Break.

   ```
   [ec2-user ~]$ CTRL+0
   ```

3. Untuk mengeluarkan SysRq perintah, tekan tombol pada keyboard Anda yang sesuai dengan perintah yang diperlukan. Misalnya, untuk menampilkan daftar SysRq perintah, tekanh.

   ```
   [ec2-user ~]$ h
   ```

   Perintah h menghasilkan sesuatu yang serupa dengan yang berikut ini.

   ```
   [ 1169.389495] sysrq: HELP : loglevel(0-9) reboot(b) crash(c) terminate-all-
   tasks(e) memory-full-oom-kill(f) kill-all-tasks(i) thaw-filesystems
   (j) sak(k) show-backtrace-all-active-cpus(l) show-memory-usage(m) nice-all-RT-
   tasks(n) poweroff(o) show-registers(p) show-all-timers(q) unraw(r
   ) sync(s) show-task-states(t) unmount(u) show-blocked-tasks(w) dump-ftrace-
   buffer(z)
   ```

SSH client

Untuk digunakan SysRq dalam klien SSH

1. Hubungkan ke konsol serial instans.
2. Untuk mengirim permintaan jeda, tekan `~B` (tilde, diikuti dengan huruf besar `B`).

```
[ec2-user ~]$ ~B
```

3. Untuk mengeluarkan SysRq perintah, tekan tombol pada keyboard Anda yang sesuai dengan perintah yang diperlukan. Misalnya, untuk menampilkan daftar SysRq perintah, tekanh.

   ```
   [ec2-user ~]$ h
   ```

   Perintah h menghasilkan sesuatu yang serupa dengan yang berikut ini.

   ```
   [ 1169.389495] sysrq: HELP : loglevel(0-9) reboot(b) crash(c) terminate-all-
   tasks(e) memory-full-oom-kill(f) kill-all-tasks(i) thaw-filesystems
   (j) sak(k) show-backtrace-all-active-cpus(l) show-memory-usage(m) nice-all-RT-
   tasks(n) poweroff(o) show-registers(p) show-all-timers(q) unraw(r
   ) sync(s) show-task-states(t) unmount(u) show-blocked-tasks(w) dump-ftrace-
   buffer(z)
   ```

   > **Note**
   >
   > Perintah yang digunakan untuk mengirim permintaan jeda mungkin berbeda bergantung pada klien SSH yang Anda gunakan.

## (Instans Windows) Gunakan SAC untuk memecahkan masalah instans Anda

Kemampuan Konsol Admin Khusus (SAC) Windows menyediakan cara untuk memecahkan masalah instans Windows. Dengan terhubung ke konsol serial instans dan menggunakan SAC, Anda dapat menginterupsi proses boot dan boot Windows dalam mode aman.

> **Note**
>
> Jika Anda mengaktifkan SAC pada instans, layanan EC2 yang mengandalkan pengambilan kata sandi tidak akan bekerja dari konsol Amazon EC2. Agen peluncuran Windows di Amazon EC2 (EC2Config, EC2Launch v1, dan EC2Launch v2) mengandalkan konsol serial untuk menjalankan berbagai tugas. Tugas-tugas ini tidak berhasil dijalankan saat Anda mengaktifkan SAC pada sebuah instans. Untuk informasi selengkapnya tentang agen peluncuran Windows di Amazon EC2, lihatthe section called “Konfigurasikan instance

Windows". Jika mengaktifkan SAC, Anda dapat menonaktifkannya nanti. Untuk informasi selengkapnya, lihat Menonaktifkan SAC dan menu boot.

Tugas

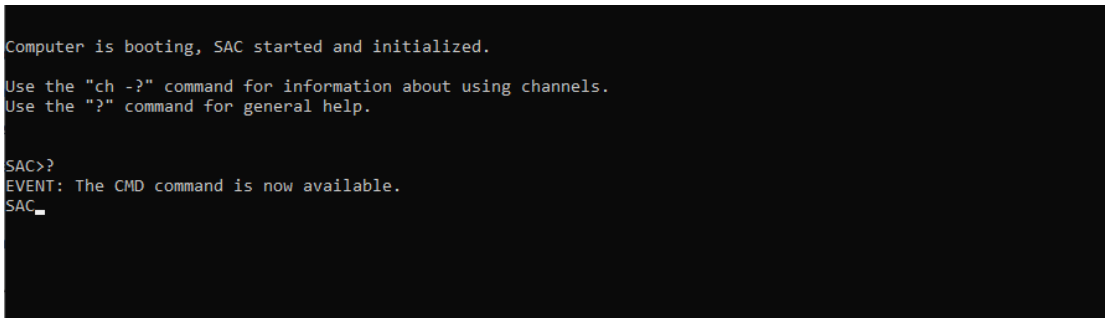
- Menggunakan SAC
- Menggunakan menu boot
- Menonaktifkan SAC dan menu boot

## Menggunakan SAC

Untuk menggunakan SAC

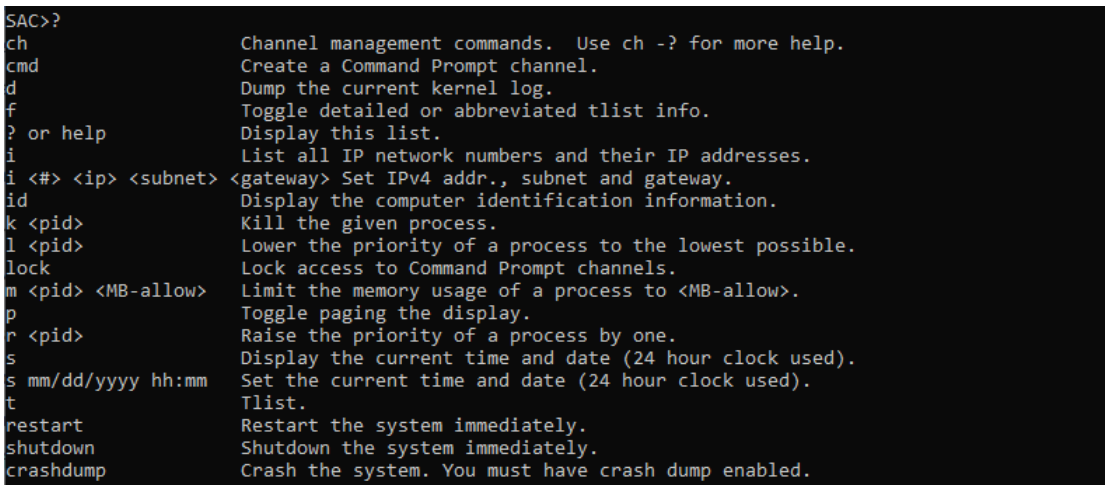
1. Hubungkan ke konsol serial.

   Jika SAC diaktifkan pada instans, konsol serial akan menampilkan perintah SAC>.



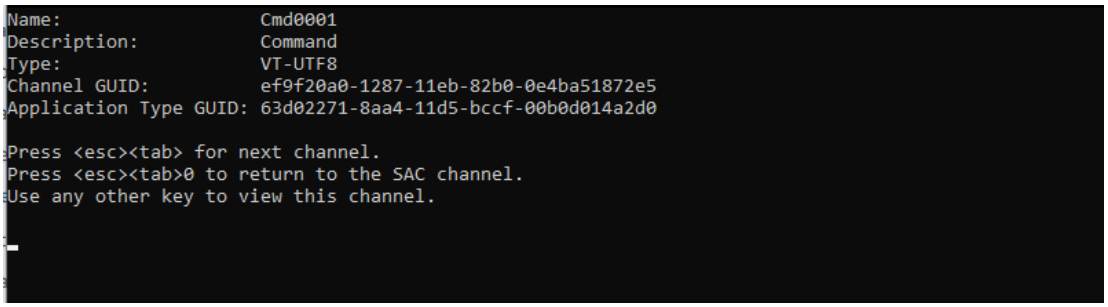

2. Untuk menampilkan perintah SAC, masukkan? , lalu tekan  Enter.

   Output yang diharapkan



3. Untuk membuat saluran prompt perintah (seperti `cmd0001` atau`cmd0002`), masukkan cmd, lalu tekan  Enter.
4. Untuk melihat saluran prompt perintah, tekan ESC, lalu tekan TAB.

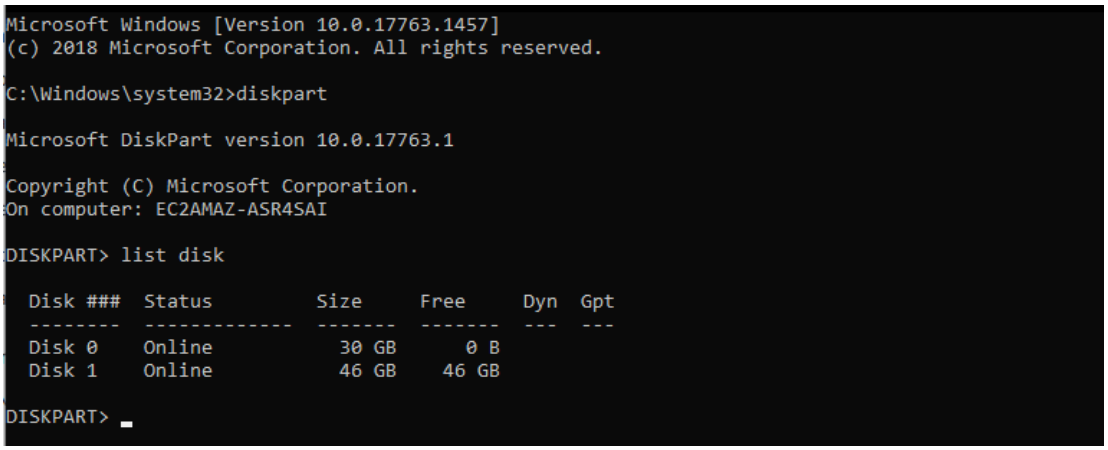
   Output yang diharapkan




5. Untuk beralih saluran, tekan ESC+TAB+nomor saluran secara bersamaan. Misalnya, untuk beralih ke saluran `cmd0002` (jika sudah dibuat), tekan ESC+TAB+2.
6. Masukkan kredensial yang diperlukan oleh saluran prompt perintah.




   Prompt perintah adalah perintah shell berfitur lengkap yang sama dengan yang Anda dapatkan di desktop, tetapi dengan pengecualian bahwa perintah tersebut tidak mengizinkan pembacaan karakter yang sudah dikeluarkan.



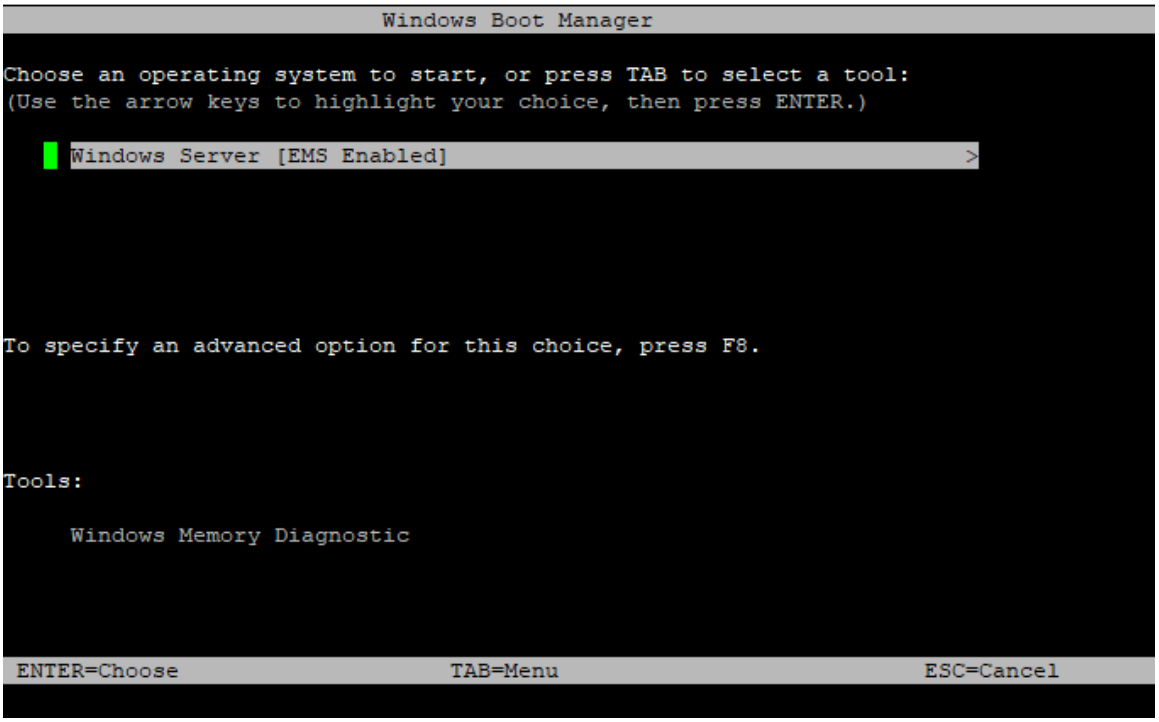

PowerShell juga dapat digunakan dari command prompt.

Perhatikan bahwa Anda mungkin perlu mengatur preferensi perkembangan ke mode diam.

```
PS C:\Windows\system32> $ProgressPreference="SilentlyContinue"
PS C:\Windows\system32> $computerInfo = Get-ComputerInfo
PS C:\Windows\system32> $computerInfo.Csprocessors[0].Name
Intel(R) Xeon(R) Platinum 8124M CPU @ 3.00GHz
PS C:\Windows\system32> $computerInfo.Csprocessors[0].Description
Intel64 Family 6 Model 85 Stepping 4
PS C:\Windows\system32>
```

### Menggunakan menu boot

Jika instance mengaktifkan menu boot dan dimulai ulang setelah terhubung melalui SSH, Anda akan melihat menu boot, sebagai berikut.




### Perintah menu boot

ENTER

Mulai entri yang dipilih dari sistem operasi.

TAB

Beralih ke menu Alat.

ESC

Membatalkan dan memulai ulang instans.

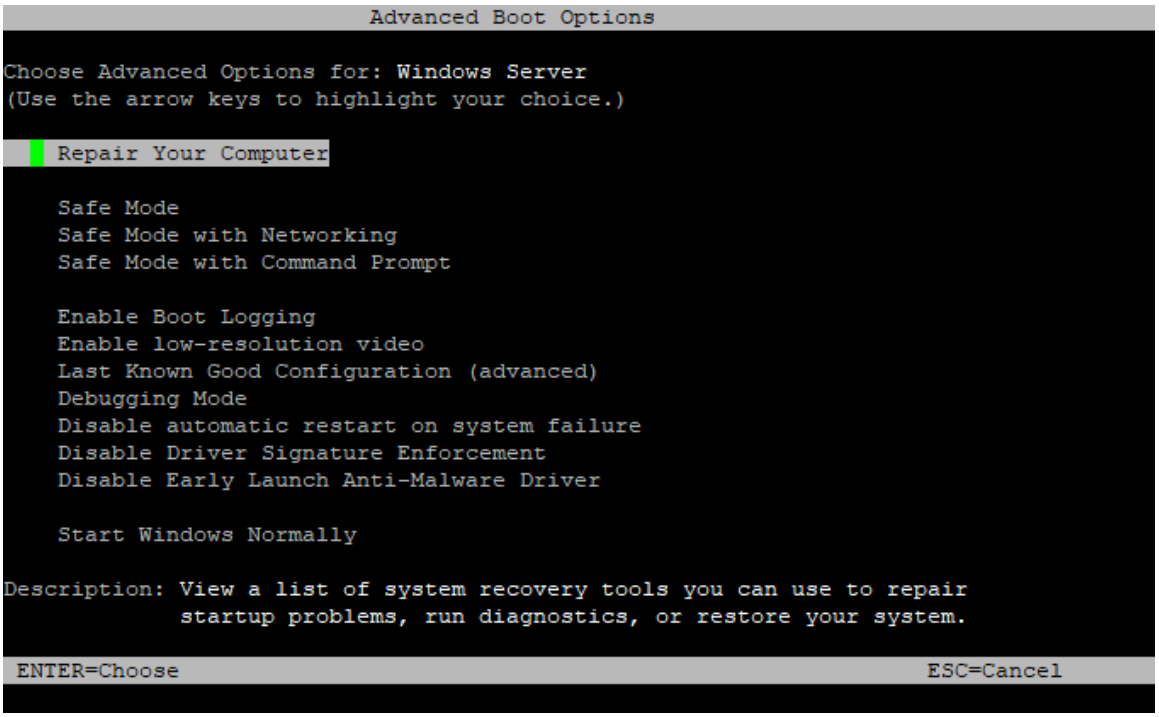
ESC diikuti dengan tombol angka 8

Sama dengan menekan F8. Menampilkan opsi lanjutan untuk item yang dipilih.

Tombol ESC + panah kiri

Kembali ke menu boot awal.

Tombol ESC tidak membawa Anda kembali ke menu utama karena Windows menunggu untuk melihat jika urutan keluar sedang berlangsung.




Menonaktifkan SAC dan menu boot

Jika mengaktifkan SAC dan menu boot, Anda dapat menonaktifkan fitur ini nanti.

Gunakan salah satu metode berikut untuk mengaktifkan SAC dan menu boot pada instans.

PowerShell

Untuk menonaktifkan SAC dan menu boot pada instans Windows

1. Hu bungkan ke instance Anda dan lakukan langkah-langkah berikut dari baris PowerShell perintah yang ditinggikan.
2. Pertama nonaktifkan menu boot dengan mengubah nilainya menjadi no.

```
bcdedit /set '{bootmgr}' displaybootmenu no
```

3. Kemudian nonaktifkan SAC dengan mengubah nilainya menjadi `off`.

```
bcdedit /ems '{current}' off
```

4. Terapkan konfigurasi yang diperbarui dengan melakukan boot ulang instans.

```
shutdown -r -t 0
```

Command prompt

Untuk menonaktifkan SAC dan menu boot pada instans Windows

1. Hubungkan ke instans Anda dan lakukan langkah-langkah berikut dari prompt perintah.
2. Pertama nonaktifkan menu boot dengan mengubah nilainya menjadi no.

```
bcdedit /set {bootmgr} displaybootmenu no
```

3. Kemudian nonaktifkan SAC dengan mengubah nilainya menjadi `off`.

```
bcdedit /ems {current} off
```

4. Terapkan konfigurasi yang diperbarui dengan melakukan boot ulang instans.

```
shutdown -r -t 0
```

# Mengirim interupsi diagnostik untuk men-debug instans Amazon EC2 yang tidak dapat dijangkau

> **Warning**
>
> Interupsi diagnostik ditujukan untuk digunakan oleh pengguna tingkat lanjut. Penggunaan yang salah dapat memengaruhi instans Anda secara negatif. Mengirimkan interupsi diagnostik ke suatu instans dapat memicu crash dan boot ulang pada instans, yang dapat menyebabkan hilangnya data.

Anda dapat mengirim interupsi diagnostik ke instans yang tidak dapat dijangkau atau tidak responsif untuk memicu panik kernel secara manual untuk instance Linux, atau kesalahan berhenti (biasanya disebut sebagai kesalahan layar biru) untuk instance Windows.

Instans Linux

Sistem operasi Linux biasanya akan mengalami crash dan boot ulang ketika kepanikan kernel terjadi. Perilaku khusus dari sistem operasi bergantung pada konfigurasinya. Kepanikan kernel juga dapat digunakan untuk menyebabkan kernel sistem operasi instans melakukan tugas, seperti membuat file dump crash. Anda kemudian dapat menggunakan informasi pada file dump crash tersebut untuk melakukan analisis akar penyebab masalah dan melakukan debug instans. Data dump crash dihasilkan secara lokal oleh sistem operasi pada instans itu sendiri.

Instans Windows

Secara umum, sistem operasi Windows mengalami crash dan melakukan boot ulang ketika terjadi kesalahan penghentian, tetapi perilaku spesifiknya bergantung pada konfigurasinya. Kesalahan penghentian juga dapat menyebabkan sistem operasi menulis informasi debug, seperti dump memori kernel, ke file. Kemudian, Anda dapat menggunakan informasi ini untuk melakukan analisis akar penyebab guna melakukan debug instans. Data dump memori dihasilkan secara lokal oleh sistem operasi pada instans itu sendiri.

Sebelum mengirimkan interupsi diagnostik ke instans Anda, kami sarankan untuk membaca dokumentasi sistem operasi, kemudian membuat perubahan konfigurasi yang diperlukan.

Daftar Isi


- Tipe instans yang didukung
- Prasyarat
- Kirimkan interupsi diagnostik


## Tipe instans yang didukung

Interupsi diagnostik didukung pada semua jenis Nitro-based instance, kecuali yang didukung oleh prosesor AWS Graviton. Untuk informasi selengkapnya, lihat instance yang dibangun di atas Sistem AWS Nitro dan AWS Gravit on.

# Prasyarat

Sebelum menggunakan interupsi diagnostik, Anda harus mengonfigurasi sistem operasi instans. Ini memastikan bahwa ia melakukan tindakan yang Anda butuhkan saat panik kernel (instance Linux) atau stop error (instans Windows) terjadi.

Instans Linux

Untuk mengonfigurasi Amazon Linux 2 atau Amazon Linux 2023 untuk menghasilkan crash dump saat terjadi kepanikan kernel

1. Menghubungkan ke instans Anda.
2. Instal kexec dan kdump.

```
[ec2-user ~]$ sudo yum install kexec-tools -y
```

3. Konfigurasikan kernel untuk mencadangkan jumlah memori yang sesuai untuk kernel sekunder. Jumlah memori yang disimpan bergantung pada total memori yang tersedia pada instans Anda. Buka file `/etc/default/grub` menggunakan editor teks pilihan Anda, temukan baris yang dimulai dengan `GRUB_CMDLINE_LINUX_DEFAULT`, lalu tambahkan parameter `crashkernel` dalam format berikut: `crashkernel=`*`memory_to_reserve`*. Misalnya, untuk mencadangkan `256MB`, modifikasi file `grub` sebagai berikut:

```
GRUB_CMDLINE_LINUX_DEFAULT="crashkernel=256M console=tty0 console=ttyS0,115200n8
 net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff
 rd.shell=0"
GRUB_TIMEOUT=0
GRUB_DISABLE_RECOVERY="true"
```

4. Simpan perubahan Anda dan tutup file `grub`.
5. Bangun kembali file konfigurasi GRUB2.

```
[ec2-user ~]$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

6. Pada instans yang berbasis prosesor Intel dan AMD, perintah `send-diagnostic-interrupt` mengirimkan non-maskable interrupt (NMI) yang tidak dikenal ke instans. Anda harus mengonfigurasi kernel agar mengalami crash saat menerima NMI yang tidak dikenal. Buka file `/etc/sysctl.conf` dengan menggunakan editor teks pilihan Anda dan tambahkan berikut ini.

```
kernel.unknown_nmi_panic=1
```

7. Boot ulang dan terhubung kembali ke instans Anda.
8. Verifikasi bahwa kernel telah dilakukan boot dengan parameter `crashkernel` yang benar.

```
$ grep crashkernel /proc/cmdline
```

Contoh output berikut mengindikasikan konfigurasi yang berhasil.

```
BOOT_IMAGE=/boot/vmlinuz-4.14.128-112.105.amzn2.x86_64 root=UUID=a1e1011e-
e38f-408e-878b-fed395b47ad6 ro crashkernel=256M console=tty0 console=ttyS0,115200n8
 net.ifnames=0 biosdevname=0 nvme_core.io_timeout=4294967295 rd.emergency=poweroff
 rd.shell=0
```

9. Verifikasi bahwa layanan kdump berjalan.

```
[ec2-user ~]$ systemctl status kdump.service
```

Contoh output berikut menunjukkan hasil jika layanan kdump berjalan.

```
kdump.service - Crash recovery kernel arming
   Loaded: loaded (/usr/lib/systemd/system/kdump.service; enabled; vendor preset:
 enabled)
   Active: active (exited) since Fri 2019-05-24 23:29:13 UTC; 22s ago
  Process: 2503 ExecStart=/usr/bin/kdumpctl start (code=exited, status=0/SUCCESS)
 Main PID: 2503 (code=exited, status=0/SUCCESS)
```

> **Note**
>
> Secara default, file dump crash disimpan ke `/var/crash/`. Untuk mengubah lokasi, modifikasi file `/etc/kdump.conf` menggunakan editor teks pilihan Anda.

Untuk mengonfigurasi SUSE Linux Enterprise, Ubuntu, atau Red Hat Enterprise Linux

Pada instans yang berbasis prosesor Intel dan AMD, perintah `send-diagnostic-interrupt` mengirimkan non-maskable interrupt (NMI) yang tidak dikenal ke instans. Anda harus mengkonfigurasi kernel agar macet ketika menerima NMI yang tidak dikenal dengan menyesuaikan

file konfigurasi untuk sistem operasi Anda. Untuk informasi tentang cara mengkonfigurasi kernel agar macet, lihat dokumentasi untuk sistem operasi Anda:

- SUSE Linux Enterprise
- Ubuntu
- Red Hat Enterprise Linux (RHEL)

Instans Windows

Untuk mengonfigurasi Windows agar menghasilkan dump memori saat terjadi kesalahan penghentian

1. Terhubung ke instans Anda.
2. Buka Panel Kontrol dan pilih Sistem, Pengaturan sistem lanjutan.
3. Dalam kotak dialog Properti Sistem, pilih tab Lanjutan.
4. Di bagian Startup and Pemulihan, pilih Pengaturan...
5. Di bagian Kegagalan sistem, konfigurasikan pengaturan sesuai kebutuhan, lalu pilih OKE.

Untuk informasi selengkapnya tentang mengonfigurasi kesalahan penghentian Windows, lihat Gambaran umum opsi file dump memori untuk Windows.

## Kirimkan interupsi diagnostik

Setelah menyelesaikan perubahan konfigurasi yang diperlukan, Anda dapat mengirim interupsi diagnostik ke instans menggunakan API AWS CLI atau Amazon EC2.

AWS CLI

Untuk mengirim interupsi diagnostik ke instance Anda

Gunakan perintah send-diagnostic-interrupce.

```
aws ec2 send-diagnostic-interrupt --instance-id i-1234567890abcdef0
```

PowerShell

Untuk mengirim interupsi diagnostik ke instance Anda

Gunakan Send-EC2DiagnosticInterrupt cmdlet.

```
Send-EC2DiagnosticInterrupt -InstanceId i-1234567890abcdef0
```

# Riwayat dokumen untuk Panduan Pengguna Amazon EC2

Tabel berikut menjelaskan penambahan penting pada Panduan Pengguna Amazon EC2 mulai tahun 2019. Panduan ini juga sering diperbarui untuk mengatasi umpan balik yang Anda kirimkan kepada kami.

| Perubahan | Deskripsi | Tanggal |
| --- | --- | --- |
| Pemeriksaan izin untuk Peluncuran Cepat EC2 | Menambahkan panduan tentang cara Amazon EC2 memvalidasi izin Anda saat Anda mengaktifkan Peluncura n Cepat EC2 dengan template peluncuran, termasuk pertimbangan keamanan untuk menggunakan `$Latest` atau `$Default` meluncurkan versi template. | Agustus 14, 2026 |
| Tampilan peta untuk Wilayah dan Zona | Halaman Wilayah dan Zona di T AWS ampilan Global sekarang menyertakan tampilan Peta yang menampilk an semua Wilayah dan Zona Anda pada peta dunia interakti f. Anda dapat beralih antara tampilan Daftar dan tampilan Peta. Preferensi tampilan Anda tetap ada di seluruh sesi. | Agustus 12, 2026 |
| Ganti volume root saat ini dengan volume EBS yang ada | Anda dapat mengganti volume root instans Amazon EC2 saat ini dengan volume Amazon EBS yang sudah Anda konfigurasi. | Juli 9, 2026 |

| | | |
|---|---|---|
| Penggantian template peluncuran tambahan untuk Armada EC2 'instan' | Jenis Armada EC2 `instant` sekarang mendukung parameter pengganti an template peluncuran tambahan berikut:`IamInstanceProfile` ,, `KeyName` dan. `MetadataOptions` | Juli 9, 2026 |
| Temukan parameter SSM untuk AMI publik | Anda dapat menemukan AWS Systems Manager parameter yang terkait dengan AMI publik dan menggunak annya untuk secara otomatis mereferensikan versi terbaru AMI tersebut. | Juli 7, 2026 |
| SEV-SNP pada Host Khusus | AMD sekarang SEV-SNP didukung pada Host Khusus, memungkinkan Anda menjalankan beban kerja komputasi rahasia Anda di server fisik yang sepenuhnya didedikasikan untuk penggunaan Anda. | Juli 1, 2026 |
| AWS Tampilan Global | EC2 Global View sekarang berganti nama menjadi AWS Global View. Anda dapat menggunakannya untuk melihat sumber daya EC2 dan sumber daya lain yang didukung AWS di semua Wilayah dalam satu tampilan. | 30 Juni 2026 |

| | | |
|---|---|---|
| Precision-time kelompok penempatan | Anda dapat menggunakan grup penempatan waktu presisi untuk menempatkan instance pada perangkat keras dengan akses langsung ke sumber waktu presisi tinggi di infrastruktur. AWS | Juni 29, 2026 |
| Gunakan tanda air dengan AMI yang Diizinkan | Anda sekarang dapat menggunakan tanda air AMI dengan AMI yang Diizinkan untuk membatasi peluncuran instance hanya ke AMI dengan filter tanda air tertentu. Parameter Allowed AMI `ImageWatermarks` yang baru memiliki filter berikut:`MaximumDaysSinceSourceImageCreated` ,`MaximumDaysSinceWatermarkCreated` ,`SourceImageRegion` , dan`WatermarkKey` . | Juni 22, 2026 |
| AWS Perangkat VMClock untuk Windows | AWS menyediakan paket driver null (tanpa fungsi) untuk Windows yang memungkinkan sistem operasi mengenali perangkat. | Juni 17, 2026 |

| | | |
|---|---|---|
| Tanda air AMI | Anda dapat menggunakan tanda air AMI untuk melacak asal dan menegakkan kebijakan tata kelola untuk AMI pribadi Anda. Tanda air tetap ada saat Anda membuat AMI dari instance, menyalin AMI, atau berbagi AMI dengan akun lain. | Juni 10, 2026 |
| Batalkan Reservasi Kapasitas | Anda dapat membatalkan Reservasi Kapasitas yang akan datang. Tergantung kapan Anda membatalkan, biaya mungkin berlaku. | Juni 2, 2026 |
| Dukungan ENA Express Cross-AZ | ENA Express sekarang mendukung lalu lintas antara instans Amazon EC2 di Zona Ketersediaan yang berbeda dalam Wilayah yang sama. | 11 Mei 2026 |
| Visibilitas sumber daya terkelola | Anda dapat mengontrol apakah sumber daya yang AWS disediakan layanan atas nama Anda akan muncul di tampilan konsol Amazon EC2 dan operasi daftar API. | April 22, 2026 |
| Memulai dengan EFA dan NIXL | Anda dapat menggunakan Elastic Fabric Adapter (EFA) dengan NIXL untuk beban kerja inferensi terpilah. | Maret 17, 2026 |

| | | |
|---|---|---|
| Tutorial: Luncurkan Armada EC2 dengan Reservasi Kapasitas yang Dapat Diinterupsi | Tutorial baru menunjukkan cara membuat Armada EC2 yang menggunakan Reservasi Kapasitas Terputus untuk membuat kapasitas cadangan yang tidak terpakai tersedia untuk beban kerja yang dapat diinterupsi. | Maret 16, 2026 |
| Blok Kapasitas di Zona Lokal | Blok Kapasitas untuk ML sekarang tersedia di Zona Lokal. | Februari 25, 2026 |
| Penegakan IMDSv2 | Anda dapat menerapkan penggunaan IMDSv2 di tingkat akun untuk masing-masing. Wilayah AWS Saat diterapkan, instance hanya dapat diluncurkan jika dikonfigurasi untuk memerlukan IMDSv2. | Februari 24, 2026 |
| Virtualisasi bersarang | Anda dapat menjalankan hypervisor, seperti Hyper-V dan KVM, di dalam instans Amazon EC2 tervirtualisasi. | Februari 16, 2026 |
| Jaringan Sekunder | Jaringan Sekunder adalah jaringan virtual yang dibuat khusus untuk kasus penggunaan jaringan khusus. Anda dapat meluncurkan instans EC2 tertentu ke VPC dan Jaringan Sekunder. | Februari 9, 2026 |

| | | |
|---|---|---|
| Berbagi Blok Kapasitas | Anda dapat berbagi Blok Kapasitas EC2 dengan orang lain Akun AWS di AWS organisasi Anda. | Februari 2, 2026 |
| Reservasi Kapasitas yang Dapat Diinterupsi | Reservasi Kapasitas Terputus membantu Anda membuat kapasitas yang tidak terpakai tersedia sementara untuk beban kerja lain dalam akun Anda. | November 21, 2025 |
| Topologi instance untuk P6-B300 instance | Anda bisa mendapatkan informasi topologi instance untuk `p6-b300.48xlarge` instance. | November 18, 2025 |
| Nenek moyang AMI | Anda sekarang dapat melacak asal AMI dengan mendapatkan daftar semua AMI leluhurnya. | 12 November 2025 |
| Topologi Reservasi Kapasitas | Anda dapat menggunakan DescribeCapacityReservationTopology API untuk menjelaskan di mana kapasitas cadangan Anda akan ditempatkan relatif satu sama lain dalam hierarki jaringan sebelum Anda meluncurkan instance. | Oktober 28, 2025 |

| | | |
|---|---|---|
| Optimalkan CPU untuk penghematan lisensi | Anda sekarang dapat mengonfigurasi CPU aktif untuk menghemat biaya lisensi untuk instans Windows dan SQL Server yang disertakan dengan lisensi. | Oktober 15, 2025 |
| Manajer Kapasitas EC2 Baru | Amazon EC2 menambahkan EC2 Capacity Manager yang memungkinkan Anda memantau, menganalisis, dan mengelola penggunaan kapasitas di seluruh On-Demand Instans, Instans Spot, dan Reservasi Kapasitas dari satu lokasi. | Oktober 15, 2025 |
| AWSEC2CapacityManagerServiceRolePolicy — kebijakan terkelola baru | Amazon EC2 menambahkan kebijakan ter AWSEC2CapacityManagerServiceRolePolicy kelola untuk memungkinkan Anda mengelola sumber daya kapasitas dan berintegrasi dengan AWS Organisasi atas nama Anda. | Oktober 15, 2025 |
| Metrik `GPUPowerUtilization` | `GPUPowerUtilization` Metrik baru untuk instans komputasi yang dipercepat. | Oktober 13, 2025 |

| | | |
|---|---|---|
| Dukungan IPv6 Endpoint Connect Instance EC2 | EC2 Instance Connect Endpoint sekarang mendukung konektivitas IPv6. Anda sekarang dapat mengonfigurasi EC2 Instance Connect Endpoints untuk mendukung dual-stack atau IPv6 hanya untuk terhubung ke instans dengan alamat IPv6. | 2 Oktober 2025 |
| AMI yang diizinkan - parameter baru | AMI yang diizinkan, fitur yang memungkinkan Anda membatasi penemuan dan penggunaan AMI di dalam Akun AWS, sekarang mendukung empat parameter baru: kode AWS Marketplace produk, nama AMI, periode maksimum sejak penghenti an AMI, dan usia maksimum untuk AMI. | 25. September 2025 |
| Statistik kinerja terperinci misalnya volume toko | Amazon EC2 menyediak an statistik kinerja resolusi tinggi real-time untuk volume penyimpanan instans NVMe yang terpasang pada instans. Nitro-based | September 16, 2025 |
| Pembaruan Host Khusus Amazon EC2 Mac | Amazon EC2 Mac Dedicated Host mendukung pemulihan otomatis Host Khusus dan pemeliharaan host berbasis reboot. | Agustus 28, 2025 |

| | | |
|---|---|---|
| Salinan AMI untuk Zona Lokal | Anda sekarang dapat menyalin EBS-backed AMI Amazon ke dan dari Zona Lokal. | Agustus 28, 2025 |
| AmazonEC2ImageReferencesAccessPolicy — kebijakan terkelola baru | Amazon EC2 menambahkan kebijakan terkel AmazonEC2ImageReferencesAccessPolicy ola untuk memberikan izin untuk memindai semua jenis sumber daya yang didukung oleh API EC2 DescribeImageReferences. | Agustus 26, 2025 |
| Laporan penggunaan AMI | Jika Anda membagikan Gambar Mesin Amazon (AMI) dengan orang lain Akun AWS — baik dengan organisasi tertentu Akun AWS, unit organisasi (OU), atau secara publik — Anda dapat melihat bagaimana AMI ini digunakan dengan membuat laporan penggunaan AMI. Laporan tersebut memberikan visibilitas ke mana Akun AWS yang menggunakan AMI Anda di instans EC2 atau template peluncuran, dan berapa banyak instans atau templat peluncuran yang mereferensikan setiap AMI. | Agustus 26, 2025 |

| | | |
|---|---|---|
| Referensi AMI | Anda dapat mengidentifikasi AWS sumber daya yang mereferensikan Gambar Mesin Amazon (AMI) tertentu, terlepas dari apakah AMI bersifat publik atau pribadi, atau siapa yang memilikin ya. Visibilitas ini membantu Anda memastikan sumber daya Anda menggunakan AMI terbaru yang sesuai. | Agustus 26, 2025 |
| Ubah Titik Akhir Hubungkan Instans EC2 | Anda sekarang dapat memodifikasi Endpoint Connect Instance EC2 yang ada menggunakan AWS CLI atau SDK. | Agustus 14, 2025 |
| Kapasitas Peningkatan berbagi reservasi | Anda sekarang dapat berbagi Reservasi Kapasitas yang Anda buat di grup penempatan cluster. | Agustus 12, 2025 |

| | | |
|---|---|---|
| Kebijakan terkelola yang diperbarui untuk EC2 Instance Connect Endpoint | Untuk mendukung modifikasi Titik Akhir Connect Instance EC2 yang ada, Amazon EC2 memperbarui `Ec2InstanceConnectEndpoint` kebijakan untuk menambahkan izin untuk menetapkan dan membatalkan penetapan alamat IPv6 dan memodifikasi grup keamanan pada antarmuka jaringan yang dibuat oleh EC2 Instance Connect Endpoint. Amazon EC2 juga memperbarui kebijakan ini untuk mengganti operator `Null` kondisi dengan operator `StringLike` kondisi. | Juli 31, 2025 |
| Topologi Instance - dukungan Wilayah tambahan | DescribeInstanceTopology API sekarang tersedia di Asia Pasifik (Melbourne). | Juli 31, 2025 |
| Berhenti paksa | Anda sekarang dapat memaksa menghentikan instance Anda. | Juli 31, 2025 |
| Topologi Instance - dukungan Wilayah tambahan | DescribeInstanceTopology API sekarang tersedia di AWS GovCloud (US-West). | Juli 24, 2025 |
| Lewati penutupan OS selama penghentian instance | Anda dapat melewati shutdown sistem operasi (OS) yang anggun saat mengakhiri instance. | Juli 23, 2025 |

| | | |
|---|---|---|
| Lewati penutupan OS selama penghentian instans | Anda dapat melewati shutdown sistem operasi (OS) yang anggun saat menghenti kan instance. | Juli 23, 2025 |
| P6-B200 Blok Kapasitas di us-east-1 | P6-B200 Blok Kapasitas sekarang tersedia di AS Timur (Virginia Utara) -`us-east-1` . | Juli 23, 2025 |
| Kebijakan terkelola yang diperbarui untuk peran terkait layanan Peluncuran Cepat EC2 | Untuk membantu mencegah sumber daya yatim piatu, Amazon EC2 memperbar ui kebijakan ini untuk menambahkan izin untuk menjelaskan volume, atribut volume, dan antarmuka jaringan, serta untuk menghapus volume dan antarmuka jaringan yang dibuat EC2 Fast Launch. | Juli 17, 2025 |
| EC2 Serial Console - dukungan Wilayah tambahan | EC2 Serial Console kini juga tersedia di Asia Pasifik (Thailand) dan Meksiko (Tengah). | Juli 15, 2025 |
| EC2 Instance Connect - dukungan Wilayah tambahan | EC2 Instance Connect sekarang juga tersedia di Asia Pasifik (Thailand) dan Meksiko (Tengah). | Juli 15, 2025 |

| | | |
|---|---|---|
| Stempel waktu paket perangkat keras | Anda dapat mengaktifkan stempel waktu paket perangkat keras pada instans EC2 Anda untuk menambahkan stempel waktu presisi nano detik 64-bit ke setiap paket jaringan yang masuk. | Juni 23, 2025 |
| Hapus snapshot selama deregistrasi AMI | Saat membatalkan pendaftaran AMI, Anda dapat secara otomatis menghapus snapshot terkait secara bersamaan. | Mei 28, 2025 |
| User-initiated reboot migrasi | Ketika instans Anda menerima pemberitahuan peristiwa reboot terjadwal EC2, instans diaktifkan secara default, misalnya migrasi saat Anda me-reboot instans sebelum acara. | Mei 20, 2025 |
| Konfigurasi SIP untuk instans Mac Amazon EC2 | Anda dapat mengonfigurasi pengaturan System Integrity Protection (SIP) untuk instans Mac x86 dan instans Mac silikon Apple. | Mei 20, 2025 |
| Delegasi kepemilikan volume akar | Anda sekarang dapat mengotomatiskan delegasi kepemilikan volume root untuk volume root Amazon EBS saat memperbarui sistem operasi pada instans Mac silikon Apple. | Mei 20, 2025 |

| | | |
|---|---|---|
| Topologi instans untuk P6-B200 instance | Anda bisa mendapatkan informasi topologi instance untuk `p6-b200.48xlarge` instance. | Mei 15, 2025 |
| Kebijakan terkelola yang diperbarui untuk peran terkait layanan Peluncuran Cepat EC2 | Memperbarui kebijakan peran terkait layanan Peluncuran Cepat EC2 dengan akses untuk membuat dan memelihara aturan EventBridge acara Amazon, dan tindakan lain untuk meningkatkan pengalaman Peluncuran Cepat EC2. Untuk informasi selengkapnya tentang izin yang diberikan oleh peran terkait layanan, lihat Izin yang diberikan oleh `EC2FastLaunchServiceRolePolicy` . | Mei 14, 2025 |
| Kebijakan terkelola yang diperbarui untuk kebijakan akses penuh EC2 Fast Launch | Memperbarui kebijakan akses penuh EC2 Fast Launch dengan izin untuk Membuat dan Mengubah operasi Template Peluncuran, untuk membuat sumber daya grup jaringan dan keamanan, mengotorisasi aturan masuk, dan menghapus sumber daya yang dibuat EC2 Fast Launch. Kebijakan ini juga dapat menjelaskan dan membuat tumpukan CloudFormation , dan menghapus tumpukan yang dibuat EC2 Fast Launch. | Mei 14, 2025 |

| | | |
|---|---|---|
| Onboarding yang disederhanakan untuk Peluncuran Cepat EC2 | Peluncuran Cepat EC2 yang diperbarui untuk menyederhanakan proses orientasi. | Mei 14, 2025 |
| AWS Config integrasi dengan AMI yang Diizinkan | Gunakan AWS Config aturan ec2-instance-launched-with-allowed-ami untuk memeriksa apakah instans yang sedang berjalan atau dihentikan diluncurkan dengan AMI yang memenuhi kriteria AMI yang Diizinkan. | Maret 11, 2025 |
| Kebijakan ter AWSEC2CapacityReservationFleetRolePolicy kelola yang diperbarui | Amazon EC2 memperbarui kebijakan ter `AWSEC2CapacityReservationFleetRolePolicy` kelola untuk menggunakan operator `ArnLike` kondisi alih-alih operator `StringLike` kondisi. | Maret 3, 2025 |
| Time-based Salinan AMI | Anda sekarang dapat meminta durasi penyelesaian untuk EBS-backed operasi penyalinan AMI untuk memastikan bahwa salinan AMI selesai dalam jangka waktu tertentu. | Februari 25, 2025 |
| AmazonEC2ReadOnlyAccessKebijakan yang diperbarui | Amazon EC2 menambahkan `GetSecurityGroupsForVpc` operasi ke AmazonEC2ReadOnlyAccess kebijakan yang ada. | Desember 27, 2024 |

| | | |
|---|---|---|
| Instans terkelola | Anda sekarang dapat melihat instans terkelola Amazon EC2 di konsol EC2. | Desember 1, 2024 |
| Kebijakan deklaratif | Anda sekarang dapat menggunakan kebijakan deklaratif untuk menerapkan pengaturan tingkat akun di beberapa Wilayah dan akun secara bersamaan. Kebijakan deklaratif didukung untuk mengonfigurasi akses konsol serial EC2, AMI yang diizinkan, default IMDS, dan memblokir pengaturan akses publik untuk VPC, AMI, dan snapshot. Untuk informasi selengkapnya, lihat Kebijakan Deklaratif di Panduan AWS Organizations Pengguna, serta dokumentasi khusus untuk setiap fitur yang didukung. | Desember 1, 2024 |
| AMI yang diizinkan | Anda sekarang dapat mengontrol penemuan dan penggunaan AMI di Amazon EC2 dengan menentukan kriteria yang harus dipenuhi oleh AMI. | Desember 1, 2024 |
| Blok Kapasitas Instan | Anda sekarang dapat memesan Blok Kapasitas untuk ML, yang dapat dimulai segera setelah 30 menit. | November 21, 2024 |

| | | |
|---|---|---|
| Future-dated Reservasi Kapasitas | Anda sekarang dapat meminta Reservasi Kapasitas untuk tanggal mendatang. | November 21, 2024 |
| Perluas Blok Kapasitas | Anda sekarang dapat memperpanjang durasi Blok Kapasitas yang ada. | November 21, 2024 |
| Blok Kapasitas 6 bulan | Anda sekarang dapat memesan Blok Kapasitas untuk ML hingga 6 bulan (182 hari). | November 21, 2024 |
| Set filter yang disimpan | Anda sekarang dapat membuat grup filter yang disesuaikan dan menggunak annya kembali untuk melihat sumber daya EC2 secara efisien. | 20 November 2024 |
| Perlindungan kinerja | Saat menggunakan pemilihan jenis instans berbasis atribut untuk Armada EC2 atau Spot Fleet, Anda sekarang dapat mengaktifkan perlindungan kinerja untuk memastikan bahwa jenis instans yang dipilih serupa atau melebihi dasar kinerja yang ditentukan. | 20 November 2024 |
| Hanya Reservasi Kapasitas | Anda sekarang dapat menentukan bahwa instance hanya akan berjalan di grup sumber daya Reservasi Kapasitas atau Reservasi Kapasitas. | 20 November 2024 |

| | | |
|---|---|---|
| Identifikasi sumber AMI | Anda sekarang dapat mengidentifikasi sumber AMI yang digunakan untuk membuat AMI. | November 13, 2024 |
| Kapasitas terpisah | Anda dapat memisahkan kapasitas dari Reservasi Kapasitas yang ada untuk membuat reservasi baru. | Oktober 30, 2024 |
| Pindahkan kapasitas | Anda sekarang dapat memindahkan kapasitas dari satu Reservasi Kapasitas ke yang lain. | Oktober 30, 2024 |
| Tutorial pemula | Dua tutorial baru untuk pemula: Luncurkan instance EC2 pertama saya dan Luncurkan instance EC2 uji dan hubungkan ke sana. | Oktober 21, 2024 |
| Dukungan Windows Server 2025 | Tambahkan dukungan untuk Windows Server 2025. | Oktober 16, 2024 |
| Tampilan Global EC2 | EC2 Global View sekarang memungkinkan Anda untuk melihat Reservasi Kapasitas dan Blok Kapasitas di akun Anda di semua Wilayah. | Oktober 16, 2024 |

| | | |
|---|---|---|
| Pemeliharaan host migrasi langsung | Amazon EC2 Dedicated Host sekarang mendukung pemeliharaan host migrasi langsung, yang secara otomatis memigrasikan instans yang didukung dari Host Khusus yang rusak ke Host Khusus pengganti tanpa menghentikan dan memulai ulang instans tersebut. | Oktober 15, 2024 |
| Penetapan penagihan untuk Reservasi Kapasitas bersama | Anda sekarang dapat menetapkan penagihan kapasitas yang tersedia dari Reservasi Kapasitas bersama ke akun konsumen milik AWS organisasi yang sama. | Oktober 14, 2024 |
| Jam perangkat keras PTP - dukungan Wilayah tambahan | Jam perangkat keras PTP sekarang juga tersedia di AS Timur (Ohio) dan Asia Pasifik (Malaysia). | September 23, 2024 |
| EC2 Instance Connect mendukung IPv6 | Anda sekarang dapat menggunakan EC2 Instance Connect untuk menyambung ke alamat IPv6 publik instans Anda. | September 23, 2024 |
| Daftar awalan EC2 Instance Connect | Anda sekarang dapat memilih daftar awalan terkelola untuk alamat IPv4 atau IPv6 saat membuat aturan di grup keamanan Anda untuk mengizinkan lalu lintas SSH dari layanan EC2 Instance Connect. | September 23, 2024 |

| | | |
|---|---|---|
| Kemampuan baru untuk mengelola Reserv On-Demand asi Kapasitas | Anda sekarang dapat membagi Reservasi Kapasitas , memindahkan kapasitas antara Reservasi Kapasitas , dan mengubah atribut kelayakan instans Reservasi Kapasitas Anda. | Agustus 14, 2024 |
| Dukungan hibernasi untuk C6g, C6gn, C6gd, C7g, C7gd, M6g, M6gd, M7g, M7gd, R6g, dan R6gd | Hibernasi instans yang baru diluncurkan yang berjalan pada tipe instans C6g, C6gn, C6gd, C7g, C7gd, M6g, M6gd, M7g, M7gd, R6g, dan R6gd. | Juli 30, 2024 |
| Dukungan hibernasi untuk AMI yang mendukung jenis instans Graviton | Hibernasi instans yang baru diluncurkan yang diluncurkan dari AMI Amazon Linux atau Ubuntu yang mendukung jenis instance Graviton. | Juli 30, 2024 |
| Jenis instans tambahan yang didukung untuk Credential Guard | Anda sekarang dapat mengaktifkan Credential Guard untuk instans C7i,, M7i C7-flex,, R7i, M7i-flex, dan T3. R7i-flex | Juni 26, 2024 |
| Pencari tipe instans EC2 — parameter tambahan | Pencari tipe instans EC2 sekarang menyediakan parameter tambahan bagi Anda untuk menentukan persyaratan yang lebih rinci untuk beban kerja Anda. | 5 Juni 2024 |

| | | |
|---|---|---|
| Kebijakan terkelola baru untuk EC2 Fast Launch | Menambahkan EC2FastLaunchFullAccess kebijakan untuk melakukan tindakan API terkait dengan fitur Peluncuran Cepat EC2 dari instance. | Mei 14, 2024 |
| Perlindungan deregistrasi AMI | Anda dapat mengaktifkan perlindungan deregistrasi pada AMI untuk mencegah penghapusan yang tidak disengaja atau berbahaya. | April 23, 2024 |
| Jam perangkat keras PTP - dukungan jenis instance | Jam perangkat keras PTP sekarang tersedia pada jenis instans M7a, M7g, M7i, R7a, dan R7i. | April 22, 2024 |
| Menambahkan pertimbangan kinerja Nitro untuk jaringan yang ditingkatkan | Halaman ini berfokus pada pertimbangan jaringan untuk membantu penyetelan kinerja instans Amazon EC2 berbasis Nitro Anda. | April 4, 2024 |
| Kebijakan terkelola baru untuk snapshot EBS berbasis VSS | Amazon EC2 VSS memiliki kebijakan terkelola IAM baru yang tersedia yang dapat Anda tambahkan ke peran profil instans untuk memastikan bahwa izin Anda tetap mutakhir dan mengikuti praktik terbaik. | Maret 28, 2024 |
| Jam perangkat keras PTP - AS Timur (Virginia Utara) | Jam perangkat keras PTP sekarang tersedia di Wilayah AS Timur (Virginia Utara). | 26 Maret 2024 |

| | | |
|---|---|---|
| Setel IMDSv2 sebagai default akun | Anda dapat mengatur semua peluncuran instans EC2 baru di akun Anda untuk menggunakan Instance Metadata Service Version 2 (IMDSv2) secara default. | Maret 25, 2024 |
| Tandai AMI Linux baru yang dibuat dari snapshot | Saat Anda membuat AMI Linux dari snapshot, Anda dapat menandai AMI baru. | Maret 7, 2024 |
| Tandai AMI dan snapshot baru saat menyalin | Saat Anda menyalin AMI, Anda dapat menandai AMI baru dan snapshot baru dengan tag yang sama, atau Anda dapat menandainya dengan tag yang berbeda. | Maret 7, 2024 |
| Hapus halaman Paket AWS Manajemen | Paket AWS Manajemen terutama digunakan dengan Windows Server 2012 dan sebelumnya. Versi platform OS lama tersebut tidak lagi didukung. Untuk mengelola dan memecahkan masalah armada server yang berjalan di dalam AWS dan di lokasi, lihat AWS Systems Manager Fleet Manager. | Februari 12, 2024 |

| | | |
|---|---|---|
| EC2 Instance Connect sudah diinstal sebelumnya pada AMI macOS | EC2 Instance Connect sekarang sudah diinstal sebelumnya di macOS Sonoma 14.2.1 atau yang lebih baru, macOS Ventura 13.6.3 atau yang lebih baru, dan macOS Monterey 12.7.2 atau AMI yang lebih baru. | Januari 26, 2024 |
| Dukungan EC2 Instance Connect untuk CentOS, macOS, dan RHEL | Anda sekarang dapat menginstal EC2 Instance Connect pada AMI CentOS, macOS, dan RHEL yang didukung. | 6 Desember 2023 |
| Dukungan hibernasi untuk C7a, C7i, R7a, R7i, dan R7iz | Hibernasikan instans yang baru diluncurkan yang berjalan pada tipe instans C7a, C7i, R7a, R7i, dan R7iz. | 1 Desember 2023 |
| Pemilih tipe instans EC2 Amazon Q | Pemilih tipe instans EC2 Amazon Q mempertim bangkan kasus penggunaa n, tipe beban kerja, dan preferensi produsen CPU, serta cara Anda mempriori taskan harga dan performa. Pemilih tipe instans EC2 Amazon Q kemudian menggunakan data ini untuk memberikan panduan dan saran untuk tipe instans Amazon EC2 yang paling cocok untuk beban kerja baru Anda. | 28 November 2023 |

| | | |
|---|---|---|
| EC2 Tingkat Gratis | Anda dapat melacak penggunaan Tingkat Gratis EC2 dari Dasbor EC2. | 26 November 2023 |
| Console-to-Code | Console-to-Code dapat membantu Anda memulai dengan kode otomatisasi Anda. Console-to-Code merekam tindakan konsol Anda, lalu menggunakan AI generatif untuk menyarankan kode dalam format kode infrastruktur pilihan Anda. Anda dapat menggunakan kode tersebut sebagai titik awal, menyesuaikannya agar siap produksi untuk kasus penggunaan khusus Anda. | 26 November 2023 |
| Batas waktu pelacakan koneksi idle yang dapat dikonfigurasi | Koneksi grup keamanan yang tetap idle dapat menyebabkan terbebaninya pelacakan koneksi dan menyebabkan koneksi tidak dilacak dan paket terputus. Anda sekarang dapat mengatur batas waktu dalam hitungan detik untuk pelacakan koneksi grup keamanan pada antarmuka jaringan Elastis. | 17 November 2023 |

| | | |
|---|---|---|
| Clock perangkat keras PTP | Instans yang didukung sekarang memiliki clock perangkat keras Precision Time Protocol (PTP). Clock perangkat keras PTP mendukung NTP atau koneksi PTP langsung. | 16 November 2023 |
| Mengubah tipe instans dari instans yang diaktifkan untuk hibernasi | Anda sekarang dapat mengubah tipe instans dari instans yang diaktifkan untuk hibernasi saat berada dalam status `stopped`. | 16 November 2023 |
| Topologi instans | Anda dapat menggunakan DescribeInstanceTopology API untuk mendeteksi lokasi instans, lalu menggunakan informasi ini untuk mengoptim alkan pekerjaan HPC dan ML dengan menjalankannya pada instans yang secara fisik lebih dekat satu sama lain. | 13 November 2023 |
| EC2 Fast Launch dukungan AMI bersama | Anda sekarang dapat mengaktifkan Peluncuran Cepat EC2 pada AMI yang dibagikan dengan Anda. Saat Anda mengaktifkan Peluncuran Cepat EC2 pada AMI bersama, snapshot yang telah disediakan sebelumnya untuk peluncuran lebih cepat dibuat di akun Anda. | 6 November 2023 |

| | | |
|---|---|---|
| Blok Kapasitas untuk ML | Sekarang Anda dapat memesan instans GPU di masa mendatang untuk mendukung beban kerja machine learning (ML) berdurasi pendek. | 31 Oktober 2023 |
| Hibernasi Instans Spot | Anda sekarang dapat melakukan hibernasi Instans Spot menggunakan pengalaman hibernasi dan keluarga instans yang sama yang saat ini tersedia untuk On-Demand Instans. | 24 Oktober 2023 |
| Pengaturan default untuk blokir akses publik untuk AMI | Blokir akses publik untuk AMI sekarang diaktifkan secara default untuk semua akun baru dan untuk akun yang sudah ada yang tidak memiliki AMI publik. | 20 Oktober 2023 |
| Tampilan Global Amazon EC2 | Tampilan Global Amazon EC2 mendukung tipe sumber daya tambahan dan opsi tampilan yang dapat dikustomisasi. | 18 Oktober 2023 |
| Dukungan hibernasi untuk Ubuntu 22.04.2 LTS (Jammy Jellyfish) | Hibernasikan instans Anda yang baru diluncurkan dari AMI Ubuntu 22.04.2 LTS (Jammy Jellyfish). | 16 Oktober 2023 |
| Menonaktifkan AMI | Anda dapat menonaktifkan AMI untuk mencegahnya digunakan untuk peluncuran instans. | 12 Oktober 2023 |

| | | |
|---|---|---|
| Pemeriksaan status EBS terlampir | Anda dapat menggunakan pemeriksaan status EBS terlampir untuk memantau apakah volume Amazon EBS yang dilampirkan ke instans dapat dijangkau. | 11 Oktober 2023 |
| Dukungan hibernasi untuk Red Hat Enterprise Linux 9 | Hibernasikan instans Anda yang baru diluncurkan dari AMI Red Hat Enterprise Linux 9. | 2 Oktober 2023 |
| Dukungan hibernasi untuk Microsoft Windows Server 2022 | Hibernasikan instans Anda yang baru diluncurkan dari Microsoft Windows Server 2022. | 2 Oktober 2023 |
| Dukungan hibernasi untuk AL2023 | Hibernasikan instans Anda yang baru diluncurkan dari AMI AL2023. | 2 Oktober 2023 |
| Memulai interupsi Instans Spot di Armada Spot | Anda dapat memilih Armada Spot di konsol Amazon EC2 dan memulai interupsi Instans Spot dalam armada sehingga Anda dapat menguji cara aplikasi di Instans Spot menangani interupsi. | 21 September 2023 |
| Blokir akses publik ke AMI | Anda dapat mengaktifkan blokir akses publik untuk AMI di tingkat akun untuk memblokir setiap upaya untuk membuat AMI Anda menjadi publik. | 12 September 2023 |

| | | |
|---|---|---|
| Dukungan hibernasi untuk M7i dan M7i-flex | Hibernasi instans yang baru diluncurkan yang berjalan pada M7i dan jenis instans. M7i-flex | 22 Agustus 2023 |
| EC2-Classic sudah tidak digunakan lagi | Dengan EC2-Classic, instans EC2 berjalan dalam satu jaringan datar yang dibagikan dengan pelanggan lain. Amazon VPC menggantikan EC2-Classic. Dengan Amazon VPC, instans Anda berjalan di cloud privat virtual (VPC) yang secara logis terisolasi dari akun AWS Anda. | 8 Agustus 2023 |
| Host Khusus | Anda dapat mengalokasikan Host Khusus pada aset perangkat keras tertentu di Outpost. | 20 Juni 2023 |
| Titik Akhir EC2 Instance Connect | Anda sekarang dapat terhubung ke instance melalui SSH atau RDP tanpa mengharuskan instance memiliki alamat IPv4 publik. | 13 Juni 2023 |
| IMDS Package Analyzer | Anda sekarang dapat menggunakan IMDS Packet Analyzer untuk mengident ifikasi sumber panggilan IMDSv1 pada instans EC2 Anda. | 1 Juni 2023 |

| | | |
|---|---|---|
| Instans bare metal Konsol Serial EC2 | Konsol serial EC2 sekarang mendukung konektivitas ke port serial instans bare metal yang dipilih. | 11 April 2023 |
| Kuota templat peluncuran | Anda sekarang dapat melihat kuota untuk templat peluncuran dan versi templat peluncuran di konsol Kuota Layanan serta dengan menggunakan CLI Kuota Layanan. | 3 April 2023 |
| Notifikasi pemanfaatan Reservasi Kapasitas | AWS Health sekarang mengirimkan pemberitahuan ketika pemanfaatan kapasitas untuk Reservasi Kapasitas di akun Anda turun di bawah 20 persen. | 3 April 2023 |
| Grup Reservasi Kapasitas | Anda sekarang dapat menambahkan Reservasi Kapasitas yang dibagikan dengan Anda ke grup Reservasi Kapasitas yang Anda miliki. | 30 Maret 2023 |
| Memodifikasi opsi metadata instans | Anda sekarang dapat menggunakan konsol Amazon EC2 untuk memodifikasi opsi metadata instans. | 20 Maret 2023 |
| In-place Pembaruan sistem operasi macOS | Anda sekarang dapat melakukan pembaruan sistem operasi Apple macOS pada instans M1 Mac. | 14 Maret 2023 |

| | | |
|---|---|---|
| UEFI diutamakan | Anda sekarang dapat membuat AMI tunggal yang mendukung mode boot Unified Extensible Firmware Interface (UEFI) dan Legacy BIOS. | 3 Maret 2023 |
| Memodifikasi AMI untuk IMDSv2 | Modifikasi AMI Anda yang sudah ada sehingga instans yang diluncurkan dari AMI memerlukan IMDSv2 secara default. | 28 Februari 2023 |
| Virtualization-based Keamanan Windows - Penjaga Kredensi | Anda dapat mengaktifkan Credential Guard, fitur Virtualization-based keamanan (VBS), pada instans Amazon EC2 yang didukung. | 31 Januari 2023 |
| Kebijakan terkelola baru untuk EC2 Instance Connect Endpoint | Menambahkan `Ec2InstanceConnectEndpoint` kebijakan, yang dilampirkan ke peran AWSServiceRoleForEC2InstanceConnect terkait layanan, untuk memungkinkan Amazon EC2 melakukan tindakan atas nama Anda saat Anda membuat Titik Akhir Connect Instance EC2. | Januari 24, 2023 |
| Alias AMI dalam templat peluncuran | Anda dapat menentukan AWS Systems Manager parameter alih-alih ID AMI di template peluncuran Anda untuk menghindari keharusan memperbarui template setiap kali ID AMI berubah. | 19 Januari 2023 |

| | | |
|---|---|---|
| Dukungan hibernasi untuk C6i, I3en, dan M6i | Hibernasikan instans yang baru diluncurkan yang berjalan pada tipe instans C6i, I3en, dan M6i. | 19 Desember 2022 |
| Pencegahan tumpang tindih | Tingkatkan kinerja beban kerja database I/O-intensive relasional Anda dan kurangi latensi tanpa berdampak negatif terhadap ketahanan data dengan pencegaha n penulisan robek, fitur penyimpanan blok. | 29 November 2022 |
| ENA Ekspres | Tingkatkan throughput dan minimalkan latensi ekor lalu lintas jaringan antara instans EC2 dengan ENA Ekspres. | 28 November 2022 |
| Salin tanda AMI | Saat Anda menyalin AMI, Anda dapat menyalin tanda AMI yang ditentukan pengguna secara bersamaan. | 18 November 2022 |
| Ukuran AMI untuk menyimpan dan memulihkan | Ukuran AMI (sebelum kompresi) yang dapat disimpan dan dipulihkan ke dan dari bucket Amazon S3 sekarang dapat mencapai hingga 5.000 GB. | 16 November 2022 |

| | | |
|---|---|---|
| priceCapacityOptimizedstrategi alokasi untuk Instans Spot | Armada Spot yang menggunakan strategi alokasi `priceCapacityOptimized` melihat harga dan kapasitas untuk memilih kolam Instans Spot yang paling kecil kemungkinannya untuk terganggu dan memiliki harga paling rendah. | 10 November 2022 |
| price-capacity-optimizedstrategi alokasi untuk Instans Spot | Armada EC2 yang menggunakan strategi alokasi `price-capacity-optimized` melihat harga dan kapasitas guna memilih kolam Instans Spot yang paling kecil kemungkinannya untuk terganggu dan memiliki harga paling rendah. | 10 November 2022 |
| Membatalkan AMI yang dibagikan dengan akun Anda | Jika AMI telah dibagikan dengan Anda Akun AWS dan Anda tidak lagi ingin AMI dibagikan dengan akun Anda, Anda dapat menghapus akun Anda dari izin peluncuran AMI. | 4 November 2022 |
| Mentransfer alamat IP Elastis | Anda sekarang dapat mentransfer alamat IP Elastic dari satu Akun AWS ke yang lain. | 31 Oktober 2022 |
| Mengganti volume root | Anda dapat mengganti volume Amazon EBS root untuk instans yang berjalan menggunakan AMI. | 27 Oktober 2022 |

| | | |
|---|---|---|
| Menghubungkan instans ke basis data secara otomatis | Gunakan fitur koneksi otomatis untuk menghubungkan satu instans EC2 atau lebih dengan cepat ke basis data RDS untuk memungkinkan lalu lintas di antaranya. | 10 Oktober 2022 |
| Kuota AMI | Kuota sekarang berlaku untuk membuat dan berbagi AMI. | 10 Oktober 2022 |
| Mengonfigurasi AMI untuk IMDSv2 | Konfigurasikan AMI Anda agar instans yang diluncurkan dari AMI memerlukan IMDSv2 secara default. | 3 Oktober 2022 |
| Memulai interupsi Instans Spot | Anda dapat memilih Instans Spot di konsol Amazon EC2 dan memulai interupsi sehingga Anda dapat menguji cara aplikasi di Instans Spot menangani interupsi. | 26 September 2022 |
| Penyedia AMI terverifikasi | Di konsol Amazon EC2, AMI publik yang dimiliki oleh Amazon atau merupakan partner Amazon terverifikasi ditandai sebagai Penyedia terverifikasi. | 22 Juli 2022 |
| Grup penempatan pada AWS Outposts | Menambahkan strategi penyebaran host untuk grup penempatan di Outpost. | 30 Juni 2022 |
| Host Khusus di AWS Outposts | Anda dapat mengalokasikan Host Khusus pada AWS Outposts. | 31 Mei 2022 |

| | | |
|---|---|---|
| Perlindungan penghentian instans | Untuk mencegah instans Anda berhenti secara tidak sengaja, Anda dapat mengaktifkan perlindungan penghentian untuk instans. | 24 Mei 2022 |
| Boot Aman UEFI | Boot Aman UEFI dibangun di atas proses boot aman lama Amazon EC2 dan memberikan pertahanan tambahan yang mendalam yang membantu pelanggan mengamankan perangkat lunak dari ancaman yang bertahan selama boot ulang. | 10 Mei 2022 |
| NitroTPM | Nitro Trusted Platform Module (NitroTPM) adalah perangkat virtual yang disediakan oleh AWS Nitro System dan sesuai dengan spesifikasi TPM 2.0. | 10 Mei 2022 |
| Peristiwa perubahan status AMI | Amazon EC2 sekarang menghasilkan peristiwa saat AMI mengubah status. Anda dapat menggunakan Amazon EventBridge untuk mendeteksi dan bereaksi terhadap peristiwa ini. | 9 Mei 2022 |
| Mendeskripsikan kunci publik | Anda dapat mengueri kunci publik dan tanggal pembuatan pasangan kunci Amazon EC2. | 28 April 2022 |

| | | |
|---|---|---|
| Membuat pasangan kunci | Anda dapat menentukan format kunci (PEM atau PPK) saat membuat pasangan kunci baru. | 28 April 2022 |
| Memasang sistem file Amazon FSx saat peluncuran | Anda dapat memasang Amazon FSx baru atau yang sudah ada untuk NetApp ONTAP atau Amazon FSx untuk sistem file OpenZFS saat peluncuran menggunakan wizard instance peluncuran baru. | 12 April 2022 |
| Wizard peluncuran instans baru | Pengalaman peluncuran yang baru dan ditingkatkan di konsol Amazon EC2, yang memberikan cara yang lebih cepat dan mudah untuk meluncurkan instans EC2. | 5 April 2022 |
| Mengusangkan AMI publik secara otomatis | Secara default, tanggal pengusangan semua AMI publik diatur ke dua tahun dari tanggal pembuatan AMI. | 31 Maret 2022 |
| Kategori metadata instans: autoscaling/target -lifecycle-state | Saat menggunakan grup Auto Scaling, Anda dapat mengakses status siklus hidup target instans dari metadata instans. | 24 Maret 2022 |
| Waktu peluncuran terakhir AMI | `lastLaunchedTime` menunjukkan waktu AMI Anda terakhir kali digunakan untuk meluncurkan instans. | 28 Februari 2022 |

| | | |
|---|---|---|
| Kunci ED25519 | Kunci ED25519 sekarang didukung untuk EC2 Instance Connect dan Konsol Serial EC2. | 20 Januari 2022 |
| Platform RHEL tambahan untuk Reservasi Kapasitas | Platform Red Hat Enterprise Linux tambahan untuk Reservasi On-Demand Kapasitas. | 11 Januari 2022 |
| Konfigurasikan AMI Windows untuk peluncuran lebih cepat | Konfigurasikan AMI Windows untuk meluncurkan instans hingga 65% lebih cepat, menggunakan snapshot yang telah disediakan sebelumnya. | 10 Januari 2022 |
| Tanda instans dalam metadata instans | Anda dapat mengakses tanda instans dari metadata instans. | 6 Januari 2022 |
| Reservasi Kapasitas dalam grup penempatan klaster | Anda dapat membuat Reservasi Kapasitas dalam grup penempatan klaster. | 6 Januari 2022 |
| launch-before-terminate Armada Spot | Armada Spot dapat mengakhiri Instans Spot yang menerima notifikasi penyeimbangan ulang setelah Instans Spot pengganti baru diluncurkan. | 4 November 2021 |
| launch-before-terminate Armada EC2 | Armada EC2 dapat mengakhiri Instans Spot yang menerima notifikasi penyeimbangan ulang setelah Instans Spot pengganti baru diluncurkan. | 4 November 2021 |

| | | |
|---|---|---|
| Membandingkan stempel waktu | Anda dapat menentukan waktu sebenarnya dari suatu peristiwa dengan membandingkan stempel waktu instans Amazon EC2 Linux Anda dengan. ClockBound | 2 November 2021 |
| Membagikan AMI dengan organisasi dan OU | Sekarang Anda dapat berbagi AMI dengan AWS sumber daya berikut: organisasi dan unit organisasi (OU). | 29 Oktober 2021 |
| Skor penempatan Spot | Dapatkan rekomendasi untuk Wil AWS ayah atau Zona Ketersediaan berdasarkan persyaratan kapasitas Spot Anda. | 27 Oktober 2021 |
| Attribute-based pemilihan jenis instance untuk Spot Fleet | Tentukan atribut yang harus dimiliki oleh instans, dan Amazon EC2 akan mengidentifikasi semua tipe instans dengan atribut tersebut. | 27 Oktober 2021 |
| Attribute-based pemilihan jenis instans untuk Armada EC2 | Tentukan atribut yang harus dimiliki oleh instans, dan Amazon EC2 akan mengidentifikasi semua tipe instans dengan atribut tersebut. | 27 Oktober 2021 |
| On-Demand Armada Reservasi Kapasitas | Anda dapat menggunakan Armada Reservasi Kapasitas untuk meluncurkan grup, atau armada, dari Reservasi Kapasitas. | 5 Oktober 2021 |

| | | |
|---|---|---|
| Dukungan hibernasi untuk Ubuntu 20.04 LTS - Focal | Hibernasikan instans Anda yang baru diluncurkan dari AMI Ubuntu 20.04 LTS - Focal. | 4 Oktober 2021 |
| Armada EC2 dan Reservasi On-Demand Kapasitas yang ditargetkan | EC2 Fleet dapat meluncurkan On-Demand Instans ke Reservasi `targeted` Kapasitas. | 22 September 2021 |
| Instans T3 pada Host Khusus | Dukungan untuk instans T3 di Host Khusus Amazon EC2. | 14 September 2021 |
| Dukungan hibernasi untuk RHEL, Fedora, dan CentOS | Hibernasikan instans Anda yang baru diluncurkan dari RHEL, Fedora, dan CentOS AMI. | 9 September 2021 |
| Tampilan Global Amazon EC2 | Amazon EC2 Global View memungkinkan Anda melihat VPC, subnet, instans, grup keamanan, dan volume di beberapa Wil AWS ayah dalam satu konsol. | 1 September 2021 |
| Dukungan hibernasi untuk C5d, M5d, dan R5d | Hibernasikan instans yang baru diluncurkan yang berjalan pada tipe instans C5d, M5d, dan R5d. | 19 Agustus 2021 |
| Pasangan kunci Amazon EC2 | Amazon EC2 sekarang mendukung kunci ED25519 pada instans Linux dan Mac. | 17 Agustus 2021 |

| | | |
|---|---|---|
| Prefiks untuk antarmuka jaringan | Anda dapat menetapkan rentang CIDR IPv4 atau IPv6 privat, baik secara otomatis atau manual, ke antarmuka jaringan. | 22 Juli 2021 |
| Jendela peristiwa | Anda dapat menentukan jendela peristiwa kustom yang berulang setiap minggu untuk peristiwa terjadwal yang melakukan boot ulang, menghentikan, atau mengakhiri instans Amazon EC2 Anda. | 15 Juli 2021 |
| ID sumber daya dan dukungan penandaan untuk aturan grup keamanan | Anda dapat merujuk ke aturan grup keamanan berdasarkan ID sumber daya. Anda dapat menambahkan tanda ke aturan grup keamanan. | 7 Juli 2021 |
| Mengusangkan AMI | Sekarang Anda dapat menentukan waktu AMI diusangkan. | 11 Juni 2021 |
| Penagihan per-detik Windows | Amazon EC2 mengenakan biaya untuk Server-based penggunaan Windows dan SQL per detik, dengan biaya minimum satu menit. | 10 Juni 2021 |
| Reservasi Kapasitas pada AWS Outposts | Sekarang Anda dapat menggunakan Pencadangan Kapasitas di AWS Outposts. | 24 Mei 2021 |
| Berbagi Reservasi Kapasitas | Anda sekarang dapat berbagi Reservasi Kapasitas yang dibuat Local Zones dan Wavelength Zones. | 24 Mei 2021 |

| | | |
|---|---|---|
| Penggantian volume root | Anda sekarang dapat menggunakan tugas pengganti an volume root untuk mengganti volume EBS guna menjalankan instans. | 22 April 2021 |
| Simpan dan pulihkan AMI menggunakan S3 | S EBS-backed impan AMI di S3 dan pulihkan dari S3 untuk mengaktifkan penyalinan AMI lintas partisi. | 6 April 2021 |
| Konsol Serial EC2 | Pecahkan masalah boot dan konektivitas jaringan dengan membuat sambungan ke port serial instans. | 30 Maret 2021 |
| Mode boot | Amazon EC2 sekarang mendukung boot UEFI pada instans AMD dan Intel-based EC2 tertentu. | 22 Maret 2021 |
| Membuat catatan DNS terbalik | Anda sekarang dapat mengatur pencarian DNS terbalik untuk alamat IP Elastis. | 3 Februari 2021 |
| Menandai AMI dan snapshot pada pembuatan AMI | Saat membuat AMI, Anda dapat menandai AMI dan snapshot dengan tanda yang sama, atau Anda dapat menandainya dengan tanda yang berbeda. | 4 Desember 2020 |
| Gunakan Amazon EventBrid ge untuk memantau peristiwa Spot Fleet | Buat EventBridge aturan yang memicu tindakan terprogra m sebagai respons terhadap perubahan dan kesalahan status Spot Fleet. | 20 November 2020 |

| | | |
|---|---|---|
| Gunakan Amazon EventBridge untuk memantau peristiwa Armada EC2 | Buat EventBridge aturan yang memicu tindakan terprogram sebagai respons terhadap perubahan dan kesalahan status Armada EC2. | 20 November 2020 |
| Menghapus armada instant | Hapus Armada EC2 tipe `instant` dan akhiri semua instans di armada dalam satu panggilan API. | 18 November 2020 |
| Dukungan hibernasi untuk T3 dan T3a | Hibernasikan instans yang baru diluncurkan yang berjalan pada tipe instans T3 dan T3a. | 17 November 2020 |
| Pembuatan Cepat Amazon EFS | Anda dapat membuat dan memasang sistem file Amazon EFS ke instans saat peluncuran dengan menggunakan Pembuatan Cepat Amazon EFS. | 9 November 2020 |
| Kategori metadata instans: events/recommendations/rebalance | Perkiraan waktu, dalam UTC, ketika notifikasi rekomendasi penyeimbangan ulang instans EC2 dipancarkan untuk instans. | 4 November 2020 |
| Rekomendasi penyeimbangan ulang instans EC2 | Sinyal yang memberi tahu Anda saat Instans Spot berada pada risiko gangguan yang tinggi. | 4 November 2020 |
| Reservasi Kapasitas di Wavelength Zones | Reservasi Kapasitas sekarang dapat dibuat dan digunakan di Wavelength Zones. | 4 November 2020 |

| | | |
|---|---|---|
| Penyeimbangan Ulang Kapasitas | Konfigurasikan Armada Spot atau Armada EC2 untuk meluncurkan Instans Spot pengganti saat Amazon EC2 memancarkan rekomendasi penyeimbangan ulang. | 4 November 2020 |
| Dukungan hibernasi untuk I3, M5ad, dan R5ad | Hibernasikan instans yang baru diluncurkan yang berjalan pada tipe instans I3, M5ad, dan R5ad. | 21 Oktober 2020 |
| Batas vCPU Instans Spot | Batas Instans Spot sekarang dikelola berdasarkan jumlah vCPU yang digunakan atau akan digunakan oleh Instans Spot Anda yang sedang berjalan sambil menunggu pemenuhan permintaan terbuka. | 1 Oktober 2020 |
| Reservasi Kapasitas di Local Zones | Reservasi Kapasitas sekarang dapat dibuat dan digunakan di Local Zones. | 30 September 2020 |
| Dukungan hibernasi untuk M5a dan R5a | Hibernasikan instans yang baru diluncurkan yang berjalan pada tipe instans M5a dan R5a. | 28 Agustus 2020 |
| Metadata instans memberikan informasi lokasi dan penempatan instans | Bidang metadata instans baru dalam kategori `placement` : Wilayah, nama grup penempatan, nomor partisi, ID host, dan ID Zona Ketersediaan. | 24 Agustus 2020 |

| | | |
|---|---|---|
| Grup Reservasi Kapasitas | Anda dapat menggunakan Grup AWS Sumber Daya untuk membuat kumpulan logis Reservasi Kapasitas , dan kemudian meluncurkan instance target ke grup tersebut. | 29 Juli 2020 |
| EC2Launch v2 | Anda dapat menggunakan EC2Launch v2 untuk menjalankan tugas selama startup instans, jika instans dihentikan dan kemudian dimulai, jika instans dimulai ulang, dan sesuai permintaan. EC2Launch v2 mendukung semua versi Windows Server dan menggantikan EC2Launch serta EC2Config. | 30 Juni 2020 |
| Membawa Alamat IPv6 Anda sendiri | Anda dapat membawa sebagian atau seluruh rentang alamat IPv6 dari jaringan lokal ke akun Anda AWS . | 21 Mei 2020 |
| Meluncurkan instans menggunakan parameter Systems Manager | Anda dapat menentukan AWS Systems Manager parameter alih-alih AMI saat Anda meluncurkan instance. | 5 Mei 2020 |
| Mengustomisasi notifikasi peristiwa terjadwal | Anda dapat mengustomisasi notifikasi peristiwa terjadwal untuk menyertakan tanda dalam notifikasi email. | 4 Mei 2020 |

| | | |
|---|---|---|
| Windows Server di Host Khusus | Anda dapat menggunakan AMI Windows Server yang disediakan oleh Amazon untuk menjalankan Windows Server di Host Khusus versi terbaru. | 7 April 2020 |
| Menghentikan dan memulai Instans Spot | Hentikan Instans Spot Anda yang didukung oleh Amazon EBS dan mulai kapan saja, alih-alih mengandalkan perilaku penghentian interupsi. | 13 Januari 2020 |
| Penandaan sumber daya | Anda dapat memberikan tanda pada gateway internet khusus egress, gateway lokal, tabel rute gateway lokal, antarmuka virtual gateway lokal, grup antarmuka virtual gateway lokal, gabungan VPC tabel rute gateway lokal, dan gabungan grup antarmuka virtual tabel rute gateway lokal. | 10 Januari 2020 |
| Menghubungkan ke instans Anda menggunakan Session Manager | Anda dapat memulai sesi Session Manager dengan instans dari konsol Amazon EC2. | 18 Desember 2019 |
| Host Khusus dan grup sumber daya host | Host Khusus sekarang dapat digunakan dengan grup sumber daya host. | 2 Desember 2019 |
| Berbagi Host Khusus | Anda sekarang dapat membagikan Host Khusus Anda di seluruh AWS akun. | 2 Desember 2019 |

| | | |
|---|---|---|
| Spesifikasi kredit default di tingkat akun | Anda dapat mengatur spesifikasi kredit default per keluarga instans kinerja yang dapat meledak di tingkat akun per Wil AWS ayah. | 25 November 2019 |
| Penemuan tipe instans | Anda dapat menemukan tipe instans yang sesuai dengan kebutuhan. | 22 November 2019 |
| Host Khusus | Anda sekarang dapat mengonfigurasi Host Khusus untuk mendukung berbagai tipe instans dalam satu keluarga instans. | 21 November 2019 |
| Layanan Metadata Instans Versi 2 | Anda dapat menggunakan Layanan Metadata Instans Versi 2, yang merupakan metode berorientasi sesi untuk meminta metadata instans. | 19 November 2019 |
| Elastic Fabric Adapter | Elastic Fabric Adapter sekarang dapat digunakan dengan Intel MPI 2019 Update 6. | 15 November 2019 |
| Dukungan hibernasi untuk instans On-Demand Windows | Anda dapat melakukan hibernasi instance On-Demand Windows. | 14 Oktober 2019 |
| Pembelian dalam antrean untuk Instans Terpesan | Anda dapat mengantrekan pembelian Instans Terpesan hingga tiga tahun sebelumnya. | 4 Oktober 2019 |

| | | |
|---|---|---|
| Interupsi diagnostik | Anda dapat mengirimkan interupsi diagnostik ke instans yang tidak dapat dijangkau atau tidak responsif untuk memicu kepanikan kernel. | 14 Agustus 2019 |
| Strategi alokasi kapasitas yang dioptimalkan | Dengan Armada EC2 atau Armada Spot, Anda dapat meluncurkan Instans Spot dari kolam Spot dengan kapasitas optimal untuk jumlah instans yang diluncurkan. | 12 Agustus 2019 |
| On-Demand Berbagi Reservasi Kapasitas | Anda sekarang dapat membagikan Reservasi Kapasitas Anda di seluruh AWS akun. | 29 Juli 2019 |
| Elastic Fabric Adapter | EFA kini mendukung Open MPI 3.1.4 dan Intel MPI 2019 Update 4. | 26 Juli 2019 |
| EC2 Instance Connect | EC2 Instans Connect adalah cara sederhana dan aman untuk terhubung ke instans Anda menggunakan Secure Shell (SSH). | 27 Juni 2019 |
| Pemulihan host | Mulai ulang instans Anda secara otomatis di host baru jika terjadi kegagalan perangkat keras yang tidak terduga pada Host Khusus. | 5 Juni 2019 |

| | | |
|---|---|---|
| Snapshot yang konsisten dengan aplikasi VSS | Ambil snapshot yang konsisten dengan aplikasi dari semua volume Amazon EBS yang dilampirkan ke instans Windows Anda menggunakan Run Command. AWS Systems Manager | 13 Mei 2019 |
| Asisten Platforming Ulang Windows ke Linux untuk Basis Data Microsoft SQL Server | Pindahkan beban kerja Microsoft SQL Server yang sudah ada dari sistem operasi Windows ke Linux. Tautan yang diperbarui mengarah ke Microsoft SQL Server di Panduan Pengguna Amazon EC2. | 8 Mei 2019 |
| Peningkatan Otomatis Windows | Lakukan peningkatan otomatis instans EC2 Windows menggunakan AWS Systems Manager. | 6 Mei 2019 |
| Elastic Fabric Adapter | Anda dapat melampirkan Elastic Fabric Adapter ke instans untuk mempercepat aplikasi Komputasi Performa Tinggi (HPC). | 29 April 2019 |

Untuk informasi tentang rilis jenis instans untuk Amazon EC2, lihat Ri wayat dokumen di Panduan Jenis Instans Amazon EC2.

## Sejarah untuk 2018 dan sebelumnya

Tabel berikut menjelaskan penambahan penting pada Panduan Pengguna Amazon EC2 pada tahun 2018 dan tahun-tahun sebelumnya.

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Grup penempatan partisi | 15-11-2015 | Grup penempatan partisi menyebarkan instans di seluruh partisi logis, sehingga memastikan bahwa instans di satu partisi tidak berbagi perangkat keras yang mendasari dengan instans yang berada di partisi lain. Untuk informasi selengkapnya, lihat Grup penempatan partisi. | 20 Desember 2018 |
| Menghibernasi instans EC2 Linux | 15-11-2015 | Anda dapat menghibernasi instans Linux jika diaktifkan untuk hibernasi dan memenuhi prasyarat hibernasi. Untuk informasi selengkapnya, lihat Hibernasi instans Amazon EC2 Anda. | 28 November 2018 |
| Akselerator Amazon Elastic Inference | 15-11-2015 | Anda dapat melampirkan akselerator Amazon EI ke instans Anda untuk menambahkan GPU-powered akselerasi guna mengurangi biaya menjalankan inferensi pembelajaran mendalam. | 28 November 2018 |
| Konsol spot merekomendasikan armada instans | 15-11-2015 | Konsol Spot merekomendasikan armada instans berdasarkan praktik terbaik Spot (diversifikasi instans) untuk memenuhi spesifikasi perangkat keras minimum (vCPU, memori, dan penyimpanan) untuk kebutuhan aplikasi Anda. Untuk informasi selengkapnya, lihat Membuat Armada Spot. | 20 November 2018 |
| Tipe permintaan Armada EC2 baru: instant | 15-11-2015 | Armada EC2 sekarang mendukung tipe permintaan baru, `instant`, yang dapat Anda gunakan untuk menyediakan kapasitas secara sinkron di seluruh tipe instans dan model pembelian. Permintaan `instant` mengembalikan instans yang diluncurkan dalam respons API, dan tidak mengambil tindakan lebih | 14 November 2018 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
|---|---|---|---|
| | | lanjut, sehingga memungkinkan Anda untuk mengontrol jika dan waktu instans diluncurkan. Untuk informasi selengkapnya, lihat Jenis permintaan Armada dan Armada Spot EC2. | |
| Informasi penghematan Spot | 15-11-2015 | Anda dapat menampilkan penghematan yang dihasilkan dari penggunaan Instans Spot untuk satu Armada Spot atau untuk semua Instans Spot. Untuk informasi selengkapnya, lihat Penghematan dari pembelian Instans Spot. | 5 November 2018 |
| Dukungan konsol untuk mengoptimalkan opsi CPU | 15-11-2015 | Saat meluncurkan instans, Anda dapat mengoptimalkan opsi CPU untuk menyesuaikan beban kerja atau kebutuhan bisnis tertentu menggunakan konsol Amazon EC2. Untuk informasi selengkapnya, lihat Opsi CPU untuk instans Amazon EC2. | 31 Oktober 2018 |
| Dukungan konsol untuk membuat templat peluncuran dari instans | 15-11-2015 | Anda dapat membuat templat peluncuran dengan instans sebagai dasar untuk templat peluncuran baru menggunakan konsol Amazon EC2. Untuk informasi selengkapnya, lihat Buat template peluncuran Amazon EC2. | 30 Oktober 2018 |
| On-Demand Reservasi Kapasitas | 15-11-2015 | Anda dapat memesan kapasitas untuk instans Amazon EC2 di Zona Ketersediaan tertentu untuk durasi waktu berapa pun. Hal ini memungkinkan Anda untuk membuat dan mengelola reservasi kapasitas secara independen dari diskon penagihan yang ditawarkan Instans Terpesan (RI). Untuk informasi selengkapnya, lihat Cadangan kapasitas komputasi dengan Reservasi Kapasitas EC2 On-Demand. | 25 Oktober 2018 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Bawa Alamat IP Anda Sendiri (BYOIP) | 15-11-2015 | Anda dapat membawa sebagian atau seluruh rentang alamat IPv4 publik Anda dari jaringan lokal ke akun Anda AWS . Setelah Anda membawa rentang alamat ke AWS, itu muncul di akun Anda sebagai kumpulan alamat. Anda dapat membuat alamat IP Elastis dari kolam alamat Anda dan menggunakannya dengan sumber daya AWS . Untuk informasi selengkapnya, lihat Bawa alamat IP Anda sendiri (BYOIP) ke Amazon EC2. | 23 Oktober 2018 |
| Dukungan tanda Host Khusus saat pembuatan dan konsol | 15-11-2015 | Anda dapat menandai Host Khusus saat pembuatan, dan dapat mengelola tanda Host Khusus menggunakan konsol Amazon EC2. Untuk informasi selengkapnya, lihat Mengalokasikan Host Khusus Amazon EC2 untuk digunakan di akun Anda. | 8 Oktober 2018 |
| Dukungan konsol untuk penskalaan terjadwal pada Armada Spot | 15-11-2015 | Meningkatkan atau mengurangi kapasitas armada saat ini berdasarkan tanggal dan waktu. Untuk informasi selengkapnya, lihat Penskalaan terjadwal: Skalakan Armada Spot sesuai jadwal. | 20 September 2018 |
| Strategi alokasi untuk Armada EC2 | 15-11-2015 | Anda dapat menentukan apakah On-Demand kapasitas dipenuhi dengan harga (harga terendah pertama) atau prioritas (prioritas tertinggi pertama). Anda dapat menentukan jumlah kolam Spot untuk mengalokasikan kapasitas Spot target. Untuk informasi selengkapnya, lihat Gunakan strategi alokasi untuk menentukan bagaimana EC2 Fleet atau Spot Fleet memenuhi Spot dan kapasitas On-Demand. | 26 Juli 2018 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Strategi alokasi untuk Armada Spot | 15-11-2015 | Anda dapat menentukan apakah On-Demand kapasitas dipenuhi dengan harga (harga terendah pertama) atau prioritas (prioritas tertinggi pertama). Anda dapat menentukan jumlah kolam Spot untuk mengalokasikan kapasitas Spot target. Untuk informasi selengkapnya, lihat Gunakan strategi alokasi untuk menentukan bagaimana EC2 Fleet atau Spot Fleet memenuhi Spot dan kapasitas On-Demand. | 26 Juli 2018 |
| Otomatiskan siklus hidup snapshot | 15-11-2015 | Anda dapat menggunakan Amazon Data Lifecycle Manager guna mengotomatisasi pembuatan dan penghapusan snapshot untuk volume EBS Anda. Untuk informasi selengkapnya, lihat Amazon Data Lifecycle Manager. | 12 Juli 2018 |
| Opsi CPU templat peluncuran | 15-11-2015 | Saat Anda membuat templat peluncuran alat baris perintah, Anda dapat mengoptimalkan opsi CPU untuk menyesuaikan beban kerja atau kebutuhan bisnis tertentu. Untuk informasi selengkapnya, lihat Buat template peluncuran Amazon EC2. | 11 Juli 2018 |
| Menandai Host Khusus | 15-11-2015 | Anda dapat menandai Host Khusus. | 3 Juli 2018 |
| Mendapatkan output konsol terbaru | 15-11-2015 | Anda dapat mengambil output konsol terbaru untuk beberapa jenis instance saat Anda menggunakan perintah get-console AWS CLI e-output. | 9 Mei 2018 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Mengoptimalkan opsi CPU | 15-11-2015 | Saat meluncurkan instans, Anda dapat mengoptimalkan opsi CPU untuk menyesuaikan beban kerja atau kebutuhan bisnis tertentu. Untuk informasi selengkapnya, lihat Opsi CPU untuk instans Amazon EC2. | 8 Mei 2018 |
| Armada EC2 | 15-11-2015 | Anda dapat menggunakan EC2 Fleet untuk meluncurkan grup instans di berbagai jenis instans EC2 dan Zona Ketersediaan, serta di seluruh model pembelian Inst On-Demand ans, Instans Cadangan, dan Instans Spot. Untuk informasi selengkapnya, lihat Armada EC2 dan Armada Spot. | 2 Mei 2018 |
| On-Demand Contoh di Spot Fleets | 15-11-2015 | Anda dapat menyertakan permintaan On-Demand kapasitas dalam permintaan Spot Fleet untuk memastikan bahwa Anda selalu memiliki kapasitas instans. Untuk informasi selengkapnya, lihat Armada EC2 dan Armada Spot. | 2 Mei 2018 |
| Menandai snapshot EBS saat pembuatan | 15-11-2015 | Anda dapat menerapkan tanda ke snapshot selama pembuatan. | 2 April 2018 |
| Mengubah grup penempatan | 15-11-2015 | Anda dapat memindahkan instans ke dalam atau ke luar grup penempatan, atau mengubah grup penempatannya. Untuk informasi selengkapnya, lihat Mengubah penempatan untuk instance EC2. | 1 Maret 2018 |
| ID sumber daya lebih panjang | 15-11-2015 | Anda dapat menggunakan format ID yang lebih panjang untuk mendapatkan lebih banyak tipe sumber daya. | 9 Februari 2018 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Peningkatan performa jaringan | 15-11-2015 | Instans di luar grup penempatan klaster sekarang dapat memperoleh keuntungan dari peningkatan bandwidth saat mengirim atau menerima lalu lintas jaringan antara instans lain atau Amazon S3. | 24 Januari 2018 |
| Menandai alamat IP Elastis | 15-11-2015 | Anda dapat menandai alamat IP Elastis. | 21 Desember 2017 |
| Layanan Amazon Time Sync | 15-11-2015 | Anda dapat menggunakan Layanan Amazon Time Sync agar waktu di instans Anda selalu akurat. Untuk informasi selengkapnya, lihat Sinkronisasi jam dan waktu presisi pada instans EC2 Anda. | 29 November 2017 |
| T2 Unlimited | 15-11-2015 | Instans T2 Unlimited dapat melonjak melampaui dasar selama yang dibutuhkan. Untuk informasi selengkapnya, lihat Instance performa yang dapat melonjak. | 29 November 2017 |
| Templat peluncuran | 15-11-2015 | Templat peluncuran dapat berisi semua atau beberapa parameter untuk meluncurkan instans, sehingga Anda tidak perlu menentukannya setiap kali meluncurkan instans. Untuk informasi selengkapnya, lihat Simpan parameter peluncuran instans di templat peluncuran Amazon EC2. | 29 November 2017 |
| Penempatan sebaran | 15-11-2015 | Grup penempatan sebaran direkomendasikan untuk aplikasi yang memiliki instans penting dalam jumlah kecil yang harus disimpan terpisah satu sama lain. Untuk informasi selengkapnya, lihat Grup penempatan tersebar. | 29 November 2017 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
|---|---|---|---|
| Hibernasi Instans Spot | 15-11-2015 | Layanan Spot dapat menghibernasi Instans Spot jika terjadi interupsi. | 28 November 2017 |
| Pelacakan target Armada Spot | 15-11-2015 | Anda dapat mengatur kebijakan penskalaan pelacakan target untuk Armada Spot. Untuk informasi selengkapnya, lihat Penskalaan pelacakan target: Menskalakan Armada Spot dengan menargetkan nilai untuk metrik tertentu. | 17 November 2017 |
| Armada Spot berintegrasi dengan Elastic Load Balancing | 15-11-2015 | Anda dapat melampirkan satu penyeimbang beban atau lebih ke Armada Spot. | 10 November 2017 |
| Menggabungkan dan memisahkan Instans Terpesan Konvertibel | 15-11-2015 | Anda dapat menukar (menggabungkan) dua atau lebih Instans Terpesan Konvertibel dengan Instans Terpesan Konvertibel yang baru. Anda juga dapat menggunakan proses modifikasi untuk memisahkan Instans Terpesan Konvertibel menjadi beberapa reservasi yang lebih kecil. Untuk informasi selengkapnya, lihat Menukar Instans Terpesan Konvertibel. | 6 November 2017 |
| Memodifikasi penghunian VPC | 15-11-2015 | Anda dapat mengubah atribut penghunian instans dari sebuah VPC dari `dedicated` menjadi `default`. Untuk informasi selengkapnya, lihat Mengubah sewa instans VPC. | 16 Oktober 2017 |
| Penagihan per detik | 15-11-2015 | Amazon EC2 mengenakan biaya untuk Linux-based penggunaan per detik, dengan biaya minimum satu menit. | 2 Oktober 2017 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Berhenti saat terjadi interupsi | 15-11-2015 | Anda dapat menentukan apakah Amazon EC2 harus menghentikan atau mengakhiri Instans Spot saat terjadi interupsi. Untuk informasi selengkapnya, lihat Perilaku interupsi Instance Spot. | 18 September 2017 |
| Menandai gateway NAT | 15-11-2015 | Anda dapat menandai gateway NAT. Untuk informasi selengkapnya, lihat Tandai sumber daya Anda. | 7 September 2017 |
| Deskripsi aturan grup keamanan | 15-11-2015 | Anda dapat menambahkan deskripsi ke aturan grup keamanan. | 31 Agustus 2017 |
| Elastic Graphics | 15-11-2015 | Lampirkan akselerator Elastic Graphics ke instans Anda untuk mempercepat performa grafis aplikasi Anda. | 29 Agustus 2017 |
| Memulihkan alamat IP Elastis | 15-11-2015 | Jika Anda merilis alamat IP Elastis untuk digunakan di VPC, Anda akan dapat memulihkannya. | 11 Agustus 2017 |
| Menandai instans Armada Spot | 15-11-2015 | Anda dapat mengonfigurasi Armada Spot untuk secara otomatis menandai instans yang diluncurkannya. | 24 Juli 2017 |
| Memberi tag pada sumber daya selama pembuatan | 15-11-2015 | Anda dapat menerapkan tanda pada instans dan volume selama pembuatan. Untuk informasi selengkapnya, lihat Tandai sumber daya Anda. Selain itu, Anda dapat menggunakan izin tingkat sumber daya berbasis tanda untuk mengontrol tanda yang diterapkan. Untuk informasi selengkapnya, lihat Berikan izin untuk menandai sumber daya Amazon EC2 selama pembuatan. | 28 Maret 2017 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
|---|---|---|---|
| Melakukan modifikasi pada volume EBS terlampir | 15-11-2015 | Dengan sebagian besar volume EBS terlampir ke sebagian besar instans EC2, Anda dapat mengubah ukuran volume, tipe, dan IOPS tanpa mencopot lampiran volume atau menghentikan instans. | 13 Februari 2017 |
| Melampirkan peran IAM | 15-11-2015 | Anda dapat melampirkan, mencopot lampiran, atau mengganti peran IAM untuk instans yang sudah ada. Untuk informasi selengkapnya, lihat IAM role untuk Amazon EC2. | 9 Februari 2017 |
| Instans Spot Khusus | 15-11-2015 | Anda dapat menjalankan Instans Spot di perangkat keras penghuni tunggal di cloud privat virtual (VPC). Untuk informasi selengkapnya, lihat Peluncuran pada perangkat keras penyewa tunggal. | 19 Januari 2017 |
| Dukungan IPv6 | 15-11-2015 | Anda dapat mengaitkan CIDR IPv6 dengan VPC dan subnet, serta menetapkan alamat IPv6 ke instans di VPC. Untuk informasi selengkapnya, lihat Pengalamatan IP instans Amazon EC2. | 1 Desember 2016 |
| Penskalaan otomatis untuk Armada Spot | | Anda sekarang dapat mengatur kebijakan penskalaan untuk Armada Spot. Untuk informasi selengkapnya, lihat Memahami penskalaan otomatis untuk Spot Fleet. | 1 September 2016 |
| Adaptor Jaringan Elastis (ENA) | 01-04-2016 | Anda sekarang dapat menggunakan ENA untuk jaringan yang ditingkatkan. Untuk informasi selengkapnya, lihat Jaringan yang disempurnakan di instans Amazon EC2. | 28 Juni 2016 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Peningkatan dukungan untuk menampilkan dan mengubah ID yang lebih panjang | 01-04-2016 | Anda sekarang dapat melihat dan memodifikasi pengaturan ID yang lebih panjang untuk pengguna IAM lain, peran IAM lain, atau pengguna root. | 23 Juni 2016 |
| Salin snapshot Amazon EBS terenkripsi antar akun AWS | 01-04-2016 | Anda sekarang dapat menyalin snapshot EBS terenkripsi antar AWS akun. | 21 Juni 2016 |
| Mengambil tangkapan layar dari konsol instans | 01-10-2015 | Anda sekarang dapat memperoleh informasi tambahan saat melakukan debugging instans yang tidak dapat dijangkau. Untuk informasi selengkapnya, lihat Mengambil tangkapan layar instans yang tidak dapat dijangkau. | 24 Mei 2016 |
| Dua tipe volume EBS baru | 01-10-2015 | Anda sekarang dapat membuat volume HDD Throughput yang Dioptimalkan (st1) dan HDD Cold (sc1). | 19 April 2016 |
| Menambahkan NetworkPacketsOut metrik NetworkPacketsIn dan baru untuk Amazon EC2 | | Menambahkan NetworkPacketsOut metrik NetworkPacketsIn dan baru untuk Amazon EC2. Untuk informasi selengkapnya, lihat Metrik instans. | 23 Maret 2016 |
| CloudWatch metrik untuk Spot Fleet | | Anda sekarang bisa mendapatkan CloudWatch metrik untuk Armada Spot Anda. Untuk informasi selengkapnya, lihat Pantau Armada EC2 atau Spot Fleet Anda menggunakan CloudWatch. | 21 Maret 2016 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Instans Terjadwal | 01-10-2015 | Instans Terpesan Terjadwal (Instans Terjadwal) memungkinkan Anda membeli reservasi kapasitas yang berulang setiap hari, minggu, atau setiap bulan, dengan waktu mulai dan durasi yang ditentukan. | 13 Januari 2016 |
| ID sumber daya lebih panjang | 01-10-2015 | Amazon EC2 secara bertahap memperkenalkan ID panjang yang lebih panjang untuk beberapa jenis sumber daya Amazon EC2 dan Amazon EBS. Selama periode keikutsertaan, Anda dapat mengaktifkan format ID yang lebih panjang untuk tipe sumber daya yang didukung. | 13 Januari 2016 |
| ClassicLink Dukungan DNS | 01-10-2015 | Anda dapat mengaktifkan dukungan ClassicLink DNS untuk VPC Anda sehingga nama host DNS yang dialamatkan antara instans tertaut dan EC2-Classic instans di VPC dipecahkan ke alamat IP pribadi dan bukan alamat IP publik. | 11 Januari 2016 |
| Host Khusus | 01-10-2015 | Host khusus Amazon EC2 adalah server fisik dengan kapasitas instans yang dikhususkan untuk penggunaan Anda. Untuk informasi selengkapnya, lihat Host Khusus Amazon EC. | 23 November 2015 |
| Durasi Instans Spot | 01-10-2015 | Anda sekarang dapat menentukan durasi untuk Instans Spot. Blok Spot tidak didukung (Januari 2023). | 6 Oktober 2015 |
| Memodifikasi permintaan Armada Spot | 01-10-2015 | Anda sekarang dapat memodifikasi kapasitas target permintaan Armada Spot. Untuk informasi selengkapnya, lihat Memodifikasi permintaan Armada Spot. | 29 September 2015 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Strategi alokasi yang terdiversifikasi Armada Spot | 15-04-2015 | Anda sekarang dapat mengalokasikan Instans Spot di banyak kolam Spot dengan satu permintaan Armada Spot. Untuk informasi selengkapnya, lihat Gunakan strategi alokasi untuk menentukan bagaimana EC2 Fleet atau Spot Fleet memenuhi Spot dan kapasitas On-Demand. | 15 September 2015 |
| Pembobotan instans Armada Spot | 15-04-2015 | Anda sekarang dapat menentukan unit kapasitas yang dikontribusikan oleh setiap tipe instans untuk performa aplikasi, dan menyesuaikan jumlah yang akan Anda bayarkan untuk Instans Spot di setiap kolam Spot yang sesuai. Untuk informasi selengkapnya, lihat Gunakan pembobotan instans untuk mengelola biaya dan kinerja Armada EC2 atau Spot Fleet. | 31 Agustus 2015 |
| Tindakan alarm boot ulang baru dan peran IAM baru untuk digunakan dengan tindakan alarm | | Menambahkan tindakan alarm boot ulang dan peran IAM baru untuk digunakan dengan tindakan alarm. Untuk informasi selengkapnya, lihat Buat alarm yang menghentikan, mengakhiri, melakukan boot ulang, atau memulihkan instans. | 23 Juli 2015 |
| Armada Spot | 15-04-2015 | Anda dapat mengelola kumpulan, atau armada, Instans Spot alih-alih mengelola permintaan Instans Spot yang terpisah. Untuk informasi selengkapnya, lihat Armada EC2 dan Armada Spot. | 18 Mei 2015 |
| Migrasikan alamat IP Elastic ke EC2-Classic | 15-04-2015 | Anda dapat memigrasikan alamat IP Elastic yang telah Anda alokasikan EC2-Classic untuk digunakan dalam VPC.. | 15 Mei 2015 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Mengimpor VM dengan banyak disk sebagai AMI | 01-03-2015 | Proses VM Import sekarang mendukung pengimporan VM dengan banyak disk sebagai AMI. Untuk informasi selengkapnya, lihat Meng impor VM sebagai Gambar Menggunakan VM Import/Export di Panduan Import/Export Pengguna VM. | 23 April 2015 |
| Systems Manager | | Systems Manager memungkinkan Anda untuk mengonfigurasi dan mengelola instans EC2. | 17 Februari 2015 |
| Systems Manager untuk Microsoft SCVMM 1.5 | | Anda sekarang dapat menggunakan Systems Manager untuk Microsoft SCVMM guna meluncurkan instans dan mengimpor VM dari SCVMM ke Amazon EC2. | 21 Januari 2015 |
| Pemulihan otomatis untuk instans EC2 | | Anda dapat membuat CloudWatch alarm Amazon yang memantau instans Amazon EC2 dan secara otomatis memulihkan instans jika menjadi rusak karena kegagalan perangkat keras yang mendasarinya atau masalah yang memerlukan AWS keterlibatan untuk diperbaiki. Instans yang dipulihkan identik dengan instans asli, termasuk ID instans, alamat IP, dan semua metadata instans.<br><br>Untuk informasi selengkapnya, lihat Pemulihan instans otomatis. | 12 Januari 2015 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| ClassicLink | 01-10-2014 | ClassicLink memungkinkan Anda untuk menautkan EC2-Classic instance Anda ke VPC di akun Anda. Anda dapat mengaitkan grup keamanan VPC dengan EC2-Classic instance, mengaktifkan komunikasi antara EC2-Classic instans dan instans di VPC menggunakan alamat IP pribadi. | 7 Januari 2015 |
| Pemberitahuan pengakhiran Instans Spot. | | Cara terbaik untuk melindungi dari interupsi Instans Spot adalah dengan merancang aplikasi Anda agar toleran terhadap kesalahan. Selain itu, Anda dapat memanfaatkan pemberitahuan pengakhiran Instans Spot, yang memberikan peringatan dua menit sebelum Amazon EC2 mengakhiri Instans Spot Anda.<br><br>Untuk informasi selengkapnya, lihat Pemberitahuan interupsi Instans Spot. | 5 Januari 2015 |
| Systems Manager untuk Microsoft SCVMM | | Manajer Sistem untuk Microsoft SCVMM menyediakan antarmuka yang sederhana dan mudah digunakan untuk mengelola AWS sumber daya, seperti instans EC2, dari Microsoft SCVMM. | 29 Oktober 2014 |
| Dukungan paginasi `DescribeVolumes` | 01-09-2014 | Panggilan API `DescribeVolumes` sekarang mendukung paginasi hasil dengan parameter `MaxResults` dan `NextToken`. Untuk informasi selengkapnya, lihat DescribeVolumes dalam Referensi API Amazon EC2. | 23 Oktober 2014 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Menambahkan dukungan untuk Amazon CloudWatch Logs | | Anda dapat menggunakan Amazon CloudWatch Logs untuk memantau, menyimpan, dan mengakses sistem, aplikasi, dan file log kustom dari instans atau sumber lain. Anda kemudian dapat mengambil data log terkait dari CloudWatch Log menggunakan CloudWatch konsol Amazon, perintah CloudWatch Logs di AWS CLI, atau CloudWatch Logs SDK. | 10 Juli 2014 |
| Halaman Batas Layanan EC2 baru | | Gunakan halaman Batas Layanan EC2 di konsol Amazon EC2 guna melihat batas saat ini untuk sumber daya yang disediakan oleh Amazon EC2 dan Amazon VPC, berdasarkan tiap wilayah. | 19 Juni 2014 |
| Volume SSD Tujuan Umum Amazon EBS | 01-05-2014 | Volume SSD Tujuan Umum menawarkan penyimpanan hemat biaya yang ideal untuk berbagai beban kerja. Volume ini memberikan latensi milidetik satu digit, kemampuan untuk meledak hingga 3.000 IOPS untuk jangka waktu yang lama, dan kinerja dasar 3. IOPS/GiB Ukuran volume SSD Tujuan Umum dapat bervariasi mulai 1 GiB hingga 1 TiB. | 16 Juni 2014 |
| AWS Paket Manajemen | | AWS Paket Manajemen sekarang mendukung untuk System Center Operations Manager 2012 R2. | 22 Mei 2014 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
|---|---|---|---|
| Enkripsi Amazon EBS | 01-05-2014 | Enkripsi Amazon EBS menawarkan enkripsi volume dan snapshot data EBS yang mulus, sehingga tidak perlu membangun dan memelihara infrastruktur manajemen kunci yang aman. Enkripsi EBS memungkinkan keamanan data diam dengan mengenkripsi data Anda menggunakan Kunci yang dikelola AWS. Enkripsi dilakukan di server yang melakukan hosting instans EC2, yang menyediakan enkripsi data saat berpindah antara instans EC2 dan penyimpanan EBS. | 21 Mei 2014 |
| Laporan Penggunaan Amazon EC2 | | Laporan Penggunaan Amazon EC2 adalah set laporan yang menunjukkan biaya dan data penggunaan EC2 Anda. | 28 Januari 2014 |
| Mengimpor mesin virtual Linux | 15-10-2013 | Proses VM Import sekarang mendukung pengimporan instans Linux. Untuk informasi selengkapnya, lihat Panduan Import/Export Pengguna VM. | 16 Desember 2013 |
| Resource-level izin untuk RunInstances | 15-10-2013 | Anda sekarang dapat membuat kebijakan AWS Identity and Access Management untuk mengontrol izin tingkat sumber daya untuk tindakan Amazon EC2 API RunInstances . Untuk informasi selengkapnya dan kebijakan contoh, lihat Manajemen identitas dan akses untuk Amazon EC2. | 20 November 2013 |
| Meluncurkan instance dari AWS Marketplace | | Anda sekarang dapat meluncurkan instance dari AWS Marketplace menggunakan panduan peluncuran Amazon EC2. Untuk informasi selengkapnya, lihat Luncurkan instans Amazon EC2 dari AMI AWS Marketplace. | 11 November 2013 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Wizard peluncuran baru | | Terdapat wizard peluncuran EC2 yang baru dan didesain ulang. Untuk informasi selengkapnya, lihat Luncurkan instans EC2 menggunakan wizard instans peluncuran di konsol. | 10 Oktober 2013 |
| Memodifikasi jenis instans Instans Cadangan | 01-10-2013 | Anda sekarang dapat memodifikasi tipe instans dari Instans Terpesan Linux dalam keluarga yang sama (misalnya, M1, M2, M3, C1). Untuk informasi selengkapnya, lihat Memodifikasi Instans Terpesan. | 9 Oktober 2013 |
| Memodifikasi Instans Terpesan Amazon EC2 | 15-08-2013 | Anda sekarang dapat memodifikasi Instans Terpesan di satu Wilayah. Untuk informasi selengkapnya, lihat Memodifikasi Instans Terpesan. | 11 September 2013 |
| Menetapkan alamat IP publik | 15-07-2013 | Anda sekarang dapat menetapkan alamat IP publik saat meluncurkan instans di VPC. Untuk informasi selengkapnya, lihat Tetapkan alamat IPv4 publik saat peluncuran. | 20 Agustus 2013 |
| Memberikan izin tingkat sumber daya | 15-06-2013 | Amazon EC2 mendukung Amazon Resource Name (ARN) dan kunci syarat baru. Untuk informasi selengkapnya, lihat Identity-based kebijakan untuk Amazon EC2. | 8 Juli 2013 |
| Salinan Snapshot Inkremental | 01-02-2013 | Anda sekarang dapat menjalankan salinan snapshot inkremental. | 11 Juni 2013 |
| AWS Paket Manajemen | | Paket AWS Manajemen menghubungkan instans Amazon EC2 dan sistem operasi Windows atau Linux yang berjalan di dalamnya. Paket AWS Manajemen adalah ekstensi untuk Microsoft System Center Operations Manager. | 8 Mei 2013 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
|---|---|---|---|
| Halaman Tanda baru | | Terdapat halaman Tanda baru di konsol Amazon EC2. Untuk informasi selengkapnya, lihat Tandai sumber daya Amazon EC2 Anda. | 4 April 2013 |
| Salin AMI dari satu Wilayah ke Wilayah lainnya | 01-02-2013 | Anda dapat menyalin AMI dari satu Wilayah ke Wilayah lainnya, memungkinkan Anda meluncurkan instans yang konsisten di lebih dari satu Wil AWS ayah dengan cepat dan mudah.<br><br>Untuk informasi selengkapnya, lihat Salin AMI Amazon EC2. | 11 Maret 2013 |
| Meluncurkan instans ke VPC default | 01-02-2013 | AWS Akun Anda dapat meluncurkan instans ke salah satu EC2-Classic atau VPC, atau hanya ke VPC, berdasarkan wilayah demi wilayah. Jika Anda dapat meluncurkan instance hanya ke VPC, VPC default dibuat untuk Anda. Saat Anda meluncurkan instance, instans diluncurkan ke VPC default Anda, kecuali Anda membuat VPC nondefault dan menentukannya saat Anda meluncurkan instance. | 11 Maret 2013 |
| Salinan snapshot EBS | 01-12-2012 | Anda dapat menggunakan salinan snapshot untuk membuat cadangan data, untuk membuat volume Amazon EBS baru, atau untuk membuat Amazon Machine Image (AMI). | 17 Desember 2012 |
| Memperbarui metrik dan pemeriksaan status EBS untuk volume IOPS SSD yang Tersedia | 01-10-2012 | Memperbarui metrik EBS untuk menyertakan dua metrik baru untuk volume IOPS SSD yang Tersedia. Selain itu juga menambahkan pemeriksaan status baru untuk volume SSD IOPS yang Tersedia. | 20 November 2012 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Status permintaan Instans Spot | 01-10-2012 | Status permintaan Instans Spot memudahkan untuk menentukan status permintaan Spot Anda. | 14 Oktober 2012 |
| Marketplace Instans Terpesan Amazon EC2 | 15-08-2012 | Marketplace Instans Terpesan mencocokkan penjual yang memiliki Instans Terpesan Amazon EC2 yang tidak lagi mereka perlukan dengan pembeli yang ingin membeli kapasitas tambahan. Instans Terpesan yang dibeli dan dijual melalui Marketplace Instans Terpesan berfungsi seperti Instans Terpesan lainnya, tetapi instans ini dapat memiliki sisa jangka waktu kurang dari standar penuh dan dapat dijual dengan harga berbeda. | 11 September 2012 |
| SSD IOPS yang Tersedia untuk Amazon EBS | 20-07-2012 | Volume SSD IOPS yang disediakan memberikan kinerja tinggi yang dapat diprediksi untuk beban kerja I/O intensif, seperti aplikasi database, yang mengandalkan waktu respons yang konsisten dan cepat. | 31 Juli 2012 |
| Peran IAM di instans Amazon EC2 | 01-06-2012 | Peran IAM untuk Amazon EC2 menyediakan:<br>• AWS kunci akses untuk aplikasi yang berjalan di instans Amazon EC2.<br>• Rotasi otomatis tombol AWS akses pada instans Amazon EC2.<br>• Izin terperinci untuk aplikasi yang berjalan di instans Amazon EC2 yang membuat permintaan ke AWS layanan Anda. | 11 Juni 2012 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Fitur Instans Spot yang memudahkan untuk memulai dan menangani potensi interupsi. | | Anda sekarang dapat mengelola Instans Spot sebagai berikut:<br>• Tentukan jumlah yang akan Anda bayarkan untuk Instans Spot menggunakan konfigurasi peluncuran Auto Scaling, dan mengatur jadwal untuk menentukan jumlah yang akan Anda bayarkan untuk Instans Spot. Untuk informasi lebih lanjut, lihat Meminta Instans Spot di Panduan Pengguna Penskalaan Otomatis Amazon EC2.<br>• Dapatkan notifikasi saat instans diluncurkan atau diakhiri.<br>• Gunakan CloudFormation templat untuk meluncurkan Instans Spot dalam tumpukan dengan AWS sumber daya. | 7 Juni 2012 |
| Mengekspor instans EC2 dan stempel waktu untuk pemeriksaan status pada Amazon EC2 | 01-05-2012 | Menambahkan dukungan untuk mengekspor instans Windows Server yang awalnya Anda impor ke EC2.<br>Menambahkan dukungan untuk stempel waktu pada status instans dan status sistem untuk menunjukkan tanggal dan waktu ketika pemeriksaan status mengalami kegagalan. | 25 Mei 2012 |
| Mengekspor instans EC2 dan stempel waktu dalam pemeriksaan status instans serta sistem untuk Amazon VPC | 01-05-2012 | Menambahkan dukungan untuk ekspor instance EC2 ke Citrix Xen, MicrosoftHyper-V, dan VMware vSphere.<br>Menambahkan dukungan untuk stempel waktu dalam instans dan pemeriksaan status sistem. | 25 Mei 2012 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
|---|---|---|---|
| AWS Marketplace AMI | 01-04-2012 | Menambahkan dukungan untuk AWS Marketplace AMI. | 19 April 2012 |
| Tingkat harga Instans Terpesan | 15-12-2011 | Menambahkan bagian baru yang berisi cara memanfaatkan harga diskon yang ada di dalam tingkatan harga Instans Terpesan. | 5 Maret 2012 |
| Antarmuka Jaringan Elastis (ENI) untuk instans EC2 di Amazon Virtual Private Cloud | 01-12-2011 | Menambahkan bagian baru tentang antarmuka jaringan elastis (ENI) untuk instans EC2 di VPC. Untuk informasi selengkapnya, lihat Antarmuka jaringan elastis. | 21 Desember 2011 |
| Tipe penawaran baru untuk Instans Terpesan Amazon EC2 | 01-11-2011 | Anda dapat memilih dari berbagai penawaran Instans Terpesan yang berisi tentang proyeksi penggunaan instans. | 1 Desember 2011 |
| Status instans Amazon EC2 | 01-11-2011 | Anda dapat melihat detail tambahan tentang status instans Anda, termasuk acara terjadwal yang direncanakan oleh AWS yang mungkin berdampak pada instans Anda. Aktivitas operasional ini mencakup boot ulang instans yang diperlukan untuk menerapkan pembaruan perangkat lunak atau patch keamanan, atau pemensiunan instans yang diperlukan jika terjadi masalah perangkat keras. Untuk informasi selengkapnya, lihat Memantau status instans Amazon EC2 Anda. | 16 November 2011 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Instans Spot di Amazon VPC | 15-07-2011 | Menambahkan informasi tentang dukungan untuk Instans Spot di Amazon VPC. Dengan pembaruan ini, pengguna dapat meluncurkan Instans Spot sebagai cloud privat virtual (VPC). Dengan meluncurkan Instans Spot di VPC, pengguna Instans Spot dapat menikmati manfaat Amazon VPC. | 11 Oktober 2011 |
| Proses VM Import untuk pengguna alat CLI disederhanakan | 15-07-2011 | Proses VM Import disederhanakan dengan fungsionalitas yang ditingkatkan dari `ImportInstance` dan `ImportVolume`, yang sekarang akan melakukan pengunggahan gambar ke Amazon EC2 setelah membuat tugas impor. Selain itu, dengan diperkenalkannya `ResumeImport`, pengguna dapat memulai kembali unggahan yang belum selesai pada titik saat tugas tersebut terhenti. | 15 September 2011 |
| Dukungan untuk mengimpor dalam format file VHD | | VM Import sekarang dapat mengimpor file citra mesin virtual dalam format VHD. Format file VHD kompatibel dengan platform Hyper-V virtualisasi Citrix Xen dan Microsoft. Dengan rilis ini, VM Import sekarang mendukung format gambar RAW, VHD dan VMDK (VMware ESX-compatible). Untuk informasi selengkapnya, lihat Panduan Import/Export Pengguna VM. | 24 Agustus 2011 |
| Pembaruan Amazon EC2 VM Import Connector untuk VMware vCenter | | Menambahkan informasi tentang versi 1.1 Amazon EC2 VM Import Connector untuk peralatan virtual VMware vCenter (Connector). Pembaruan ini mencakup dukungan proksi untuk akses Internet, penanganan kesalahan yang lebih baik, akurasi bilah kemajuan tugas yang lebih baik, dan beberapa perbaikan bug. | 27 Juni 2011 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Perubahan harga Zona Ketersediaan Instans Spot | 15-05-2011 | Menambahkan informasi tentang fitur harga Zona Ketersediaan Instans Spot. Dalam rilis ini, opsi harga Zona Ketersediaan baru telah ditambahkan sebagai bagian dari informasi yang dikembalikan saat Anda menanyakan permintaan Instans Spot dan riwayat harga Spot. Tambahan ini memudahkan penentuan harga yang diperlukan untuk meluncurkan Instans Spot ke dalam Zona Ketersediaan tertentu. | 26 Mei 2011 |
| AWS Identity and Access Management | | Menambahkan informasi tentang AWS Identity and Access Management (IAM), yang memungkinkan pengguna menentukan tindakan Amazon EC2 mana yang dapat digunakan pengguna dengan sumber daya Amazon EC2 secara umum. Untuk informasi selengkapnya, lihat Manajemen identitas dan akses untuk Amazon EC2. | 26 April 2011 |
| Instans Khusus | | Diluncurkan dalam Amazon Virtual Private Cloud (Amazon VPC) Anda, Instans Khusus adalah instans yang secara fisik diisolasi di tingkat perangkat keras host. Instans Khusus memungkinkan Anda memanfaatkan Amazon VPC dan AWS cloud, dengan manfaat termasuk penyediaan elastis sesuai permintaan dan membayar hanya untuk apa yang Anda gunakan, sambil mengisolasi instans komputasi Amazon EC2 Anda di tingkat perangkat keras. Untuk informasi selengkapnya, lihat Instans Khusus Amazon EC2. | 27 Maret 2011 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Pembaruan Instans Cadangan ke Konsol AWS Manajemen | | Pembaruan pada AWS Management Console memudahkan pengguna untuk melihat Instans Cadangan mereka dan membeli Instans Cadangan tambahan, termasuk Instans Cadangan Khusus. | 27 Maret 2011 |
| Informasi metadata | 01-01-2011 | Menambahkan informasi tentang metadata untuk merefleksikan perubahan dalam rilis 01-01-2011. Untuk informasi lebih lanjut, lihat Menggunakan metadata instans untuk mengelola instans EC2 Anda dan Kategori metadata instans. | 11 Maret 2011 |
| Amazon EC2 VM Import Connector untuk VMware vCenter | | Menambahkan informasi tentang Amazon EC2 VM Import Connector untuk peralatan virtual VMware vCenter (Connector). Connector adalah plug-in untuk VMware vCenter yang berintegrasi dengan Klien VMware vSphere dan menyediakan antarmuka pengguna grafis yang dapat Anda gunakan untuk mengimpor mesin virtual VMware ke Amazon EC2. | 3 Maret 2011 |
| Memaksa pelepasan lampiran volume | | Anda sekarang dapat menggunakan Konsol Manajemen AWS untuk memaksa pelepasan volume Amazon EBS dari instance. | 23 Februari 2011 |
| Perlindungan pengakhiran instans | | Anda sekarang dapat menggunakan Konsol AWS Manajemen untuk mencegah instans dihentikan. Untuk informasi selengkapnya, lihat Ubah perlindungan penghentian instans. | 23 Februari 2011 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| VM Import | 15-11-2010 | Menambahkan informasi tentang VM Import, yang memungkinkan Anda mengimpor mesin virtual atau volume ke Amazon EC2. Untuk informasi selengkapnya, lihat Panduan Import/Export Pengguna VM. | 15 Desember 2010 |
| Pemantauan dasar untuk instans | 31-08-2010 | Menambahkan informasi tentang pemantauan dasar untuk instans EC2. | 12 Desember 2010 |
| Filter dan Tanda | 31-08-2010 | Menambahkan informasi tentang membuat daftar, memfilter, dan menandai sumber daya. Untuk informasi lebih lanjut, lihat Temukan sumber daya Amazon EC2 Anda dan Tandai sumber daya Amazon EC2 Anda. | 19 September 2010 |
| Peluncuran Instans Idempotensi | 31-08-2010 | Menambahkan informasi tentang memastikan idempotensi saat menjalankan instans. | 19 September 2010 |
| AWS Identity and Access Management untuk Amazon EC2 | | Amazon EC2 sekarang terintegrasi dengan AWS Identity and Access Management (IAM). Untuk informasi selengkapnya, lihat Manajemen identitas dan akses untuk Amazon EC2. | 2 September 2010 |
| Penetapan Alamat IP Amazon VPC | 15-06-2010 | Pengguna Amazon VPC sekarang dapat menentukan alamat IP untuk menetapkan instans yang diluncurkan di VPC. | 12 Juli 2010 |
| Pem CloudWatch antauan Amazon untuk Volume Amazon EBS | | Pem CloudWatch antauan Amazon sekarang tersedia secara otomatis untuk volume Amazon EBS. | 14 Juni 2010 |

| Fitur | Versi API | Deskripsi | Tanggal rilis |
| --- | --- | --- | --- |
| Instans Terpesan dengan Windows | | Amazon EC2 sekarang mendukung Instans Terpesan dengan Windows. | 22 Februari 2010 |

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.